2019

MAC-PHY Frameworks For LTE And WiFi Networks' Coexistence Over The Unlicensed Band

Hadi Ahmad Kasasbeh
University of Mississippi

MAC-PHY Frameworks For LTE And WiFi Networks' Coexistence Over The Unlicensed Band

A Dissertation
presented in partial fulfillment of requirements
for the degree of Doctor of Philosophy
in the Electrical Engineering
The University of Mississippi

by

Hadi Ahmad Kasasbeh

May 2019

ABSTRACT

The fifth generation of cellular communications (5G) introduces the idea of aggregating the licensed and unlicensed spectra to increase spectrum efficiency and provide better coverage. This idea is very promising but introduces challenging issues. Some of these issues are to handle coexistence between long term evolution (LTE) and existing technologies (e.g., WiFi) on the unlicensed spectrum, and to provide access fairness among these technologies.

The main focus of this dissertation is to address these issues and to analyze the interactions between LTE and WiFi coexisting on the unlicensed spectrum. This can be done by providing some improvements in the first two communication layers in both technologies. Regarding the physical (PHY) layer, efficient spectrum sensing and data fusion techniques that consider correlated spectrum sensing readings at the LTE/WiFi users (sensors[1]) are needed. Failure to consider such correlation has been a major shortcoming of the literature. This resulted in poorly performing spectrum sensing systems if such correlation is not considered in correlated-measurements environments.

In the 5GHz band, for example, the cell size is relatively small, as a result, the sensors are in close proximity to one another, and hence, there is a high probability for their measurements to be correlated. Such correlation is considered in this dissertation in order to enhance the spectrum sensing abilities of LTE and WiFi systems.

On the Medium Access Control (MAC) layer, two challenges must be met, first, choosing the channel that satisfies the user requirements, and second, provide fairness among the users in the same system and the fairness among users of different systems. To address these two challenges simultaneously, a new channel assignment mechanism is introduced for

1 In this work, "sensor" may refer to an LTE-cell user or a WiFi-cell user, while a "Fusion Center" may refer to an LTE-cell eNB or a WiFi-cell access point. A "primary radio user" can refer to an LTE device in a WiFi-cell or vice versa.

both systems with some differences to account for the two systems being different. The main idea of this channel assignment is to choose the channel that is just good enough for the required transmission by the cell (LTE or WiFi) and leave the over-qualified channels for other cells that may need them for their transmissions. Hence, higher throughput and fairness values can be achieved simultaneously compared to ordinary techniques, such as greedy and random channel assignment mechanisms.

DEDICATION

To the dearest people in my life, my mother and father, who devote their lives to my success and progress in life.

To my beloved brother and sisters who are always in the front line of support pushing me forward with their love and genuine advice.

To my old and new friends who are always ready to hear my thoughts and share theirs.

To my dearest professors, Prof. Lei Cao and Prof. Ramanarayanan Viswanathan, who believed in me when I had nothing and supported me until I had something.

To Ole Miss for providing me with the opportunity to pursue my higher education dreams.

To all students all over the world who are trying to serve the science for a brighter future.

ACKNOWLEDGEMENTS

"Seek the wisdom that will untie your knot. Seek the path that demands your whole being."_ Rumi

I would like to thank and express my highest gratitude to my professors and dissertation advisors, Prof. Lei Cao and Prof. Ramanarayanan Viswanathan. Without them, this work would have never seen the light. They were always guiding me throughout my PhD study, shaping my ideas and thoughts, and providing me with their priceless ideas and thoughts. They are, without a doubt, my role models in research, leadership, and career ethics.

I would also like to thank my dissertation defense committee, Prof. Lei Cao, Prof. Ramanarayanan Viswanathan, Prof. John Daigle, and Dr Feng Wang for their time and thoughtful critiques that shaped this work in the best way possible.

I would like to thank all of my professors at Ole Miss who shed the light for me in many areas in science and pushed me to acquire important tools in research and life.

I would also like to thank the department of electrical engineering and the graduate school at the University of Mississippi, NASA, and Cspire for their generous grants to help students like me pursuing their higher education dreams and give them the opportunity to serve the science and research for the betterment of the humankind.

Finally, I must express my very profound gratitude to the University of Mississippi for having me as a member of Ole Miss family, and for providing the highest educational standards and facilities for high-quality graduates and society members.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Hard-Decision-Based Distributed Detection in Multi-Sensor System over Noise Correlated Sensing Channels

In spite of the importance in spectrum sensing, only a small part of the literature deals with the case of dependency among sensors' observations. In this chapter, Bayes criterion for minimizing the probability of error (P_e) is used to find the distributed decision rule at each sensor in a multi-sensor system for a binary hypothesis testing problem while considering the correlation over the sensing channels. In this chapter we propose a new, simple, yet efficient algorithm to find the local decision rules to achieve the objective of minimizing the P_e. Upon receiving the signal, each sensor sends a local hard decision to the fusion center (FC). The FC then generates a global decision using the K-out-of-N majority rule. The derived problem formulation and the proposed algorithm can be used with any joint probability density functions and with any odd number of sensors. The proposed framework also accounts for the path attenuation effect on the global decision. The results show the effectiveness and validity of the proposed framework in practical sensor systems.

1.1 Introduction

For the shared use of spectrum, spectrum sensing (SS) plays a vital role in identifying spectrum availabilities and preventing interference and collision. In wireless telecommunications area and especially in cognitive radio (CR) networks, SS is the practical and the most used way to specify which frequency channels are being used by primary radio (PR) users, hence, finding spectrum availabilities (a.k.a spectrum holes, and spectrum opportunities) for CR transmissions [39]. SS can be done using one sensor or multiple sensors (cooperative SS). Cooperative SS is very important compared to the use of one sensor because it can

enhance the performance of sensing system by exploiting the advantage of spatial diversity ([64, 22, 11]). There are two main types of methods for making detection (decision) in the cooperative SS systems: centralized detection (CD) (e.g. [72, 63, 70, 65]) and distributed (decentralized) detection (DD) (e.g. [67, 5, 76, 71, 18, 11]).

What makes DD more appealing in wireless communication is the property of being bandwidth efficient as compared to CD. It can be noticed that the SS literature suffers from the scarcity of considering correlation among the sensed signals at the sensors. In this work this correlation is considered in a DD SS system.

1.1.1 Related Work

Because of its simplicity, the vast majority of the literature deals with the case of having independent sensor measurements. The papers [67] and [5] considered the case of multilevel hypothesis testing in a DD system with independent sensed signals at the sensors. In [11], the authors considered the case of having independent observations in a hard decision (HD) based DD CR network. The authors proposed a near optimal log-likelihood ratio based spectrum sensing scheme to reduce the cooperation overhead. In [76], the case of DD of binary hypotheses was addressed. The authors concerned about the case of having two sensors with correlated sensed signals. In the same paper, the authors studied three types of fusion rules at the FC, the AND, OR, and X-OR fusion rules, in terms of the P_e for multivariate Gaussian joint probability density function (pdf) under both hypotheses. To find the decision intervals at each sensor in a two-sensor system, the authors used the idea of Gauss-Seidel iterative method. In [71] and [50], the authors tried to address the problem of having correlated sensed data at the sensors, but they built their work on the assumption of having identically distributed sensed signals at all sensors. This assumption can be considered as a special case from the general picture of correlated sensed signals and may not exist in reality. In [50], the authors took in consideration the effect of path attenuation while assuming identical marginal distributions for the sensors' observations.

This assumption also can be considered as a special case from practical scenarios.

For the general case to be considered in this paper, the authors in [40] has proposed the guidelines on how to generate the problem formulation for the system model. They also proposed the general guidelines for how to use the person-by-person algorithm to solve this problem (i.e., find the optimal sensor decision rules and FC decision rule to minimize P_e). However, no numerical result was presented in [40].

1.1.2 Motivation and Contributions

Motivated by the need for realistic sensor system, this chapter proposes a general realistic framework for the case of dependent noise random variables (RVs) in a sensor system with any odd number of sensors and using any kind of joint pdfs. The adopted decision criterion is based on HD where each sensor sends one bit as a local decision to the FC ([7, 73, 15]). The FC then generates a global decision to decide between two hypotheses. The proposed problem formulation in this paper was generated using the guidelines proposed in [40]. In this chapter, we propose a new algorithm to solve this problem. This algorithm is termed in the sequel as GSSA (Gauss-Seidel Sampling Algorithm).

GSSA consists of two overlapped phases. The first phase is the known Gauss-Seidel iterative method used to solve a multivariate system. The second phase is a new algorithm called the Sampling algorithm. The Sampling algorithm is used to find all the roots of a nonlinear function within a given range. Hence, this is very helpful in finding all the thresholds on the measurement space that divide the decision intervals at each sensor. It's worth mentioning that in a SS system with independent observations, there is only one threshold that divides two continuous decision intervals at each sensor while assuming two hypothesis [67]. On the other hand, this is not the case with the dependent RVs, where there could be more than one threshold dividing two kinds of discontinuous intervals (assuming two hypotheses), [76]. The proposed framework also takes into consideration the effect of the path attenuation (between the PR and each sensor) on the global decision performance

in terms of the P_e. This framework addresses the problem of correlated sensed data with a high level of generalization. This is because of no restriction on the number of sensors, no restriction on the joint pdf type of the correlated RVs, no assumption on the noise RVs at different sensors to be identically distributed, and the consideration of path attenuation effect on the decision. This generalization makes this framework more feasible to be implemented in a real life sensor system. It is worth mentioning that most of the previous works either consider a large number of sensor, but with less realistic assumptions (e.g. [67, 5, 11, 71, 50]), or consider high level of generality but with at most two sensors in the system (e.g. [76]).

1.2 System Model and Problem Statement

The adopted system model in this work is shown in Figure 1.1. A SS system of N sensors $\{s_1,\ s_2,\ ...,\ s_N\}$ sense whether a PR user (i.e., the user who owns the frequency channel under consideration) is using the frequency channel or not. The sensing channels (the physical channels between the sensors and the PR user) are assumed to be Additive Noise (AN) channels.

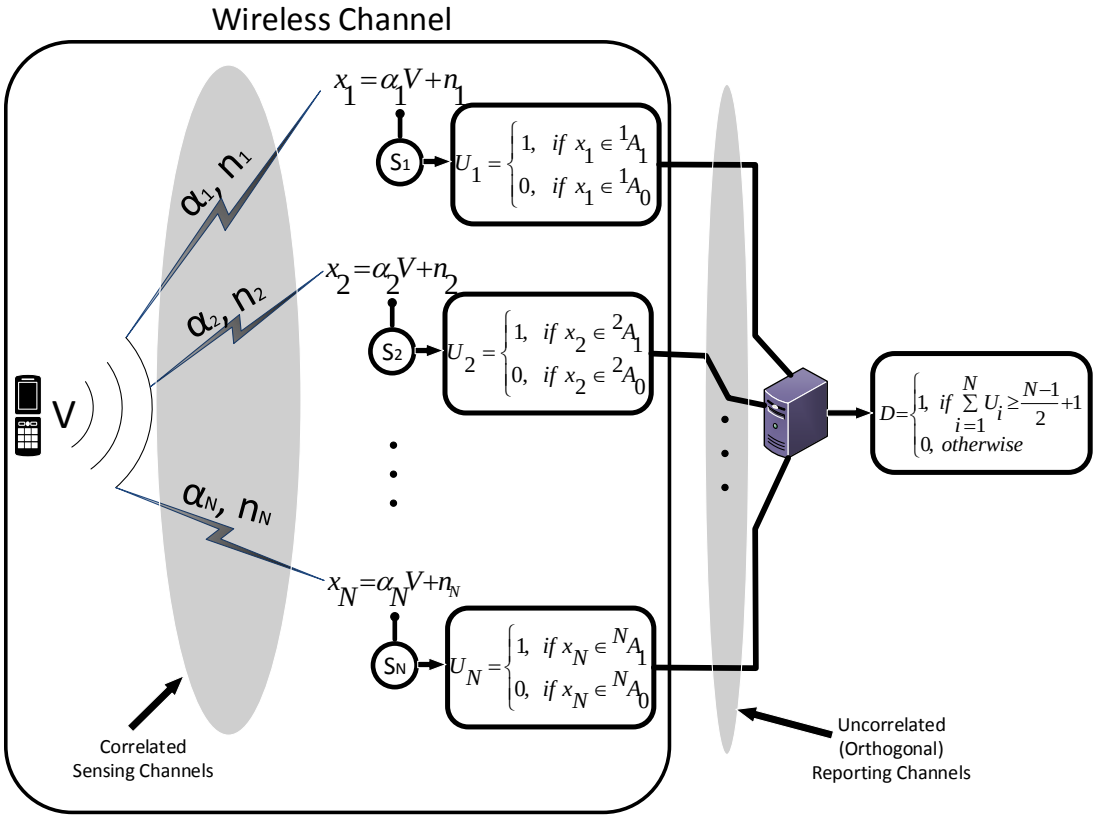


Figure 1.1. DD in multi-sensor system over correlated sensing Channels.

The noise RVs over the sensing channels are correlated with a given joint pdf that governs the correlation relation. Because of this assumption, the sensed signals by the sensors are also correlated with one joint pdf derived from the noise joint pdf under each hypothesis. The hypotheses to be tested are the existence of the PR (hypothesis K) over the channel of concern, and the absence of this PR user (hypothesis H). The joint pdfs under both hypotheses 'K' and 'H', can be denoted by f_K and f_H, respectively. The signal at each sensor not only has the added noise disturbance but also has been attenuated compared to the original signal level sent by the PR (V). As Figure 1.1 shows, each sensing channel i has its own path attenuation denoted by α_i, $i = 1, 2, ..., N$, where $0 \leq \alpha_i \leq 1$ with $\alpha_i = 0$ meaning complete attenuation and $\alpha_i = 1$ meaning no attenuation. After the signal sensing phase, each sensor quantizes its sensed signal $x_i = \alpha_i V + n_i$, $i = 1, 2, ..., N$, into one of two

levels, '1' (i.e., the existence of the PR: 'K') or '0' (i.e., the absence of the PR: 'H') according to the quantization rule U_i, $i = 1, 2, ..., N$. The quantization rule U_i decides '1' if x_i falls in the interval ${}^iA_1$, and decides '0' if x_i falls in ${}^iA_0 = \overline{{}^iA_1}$. Hence, U_i can be written as:

$$U_i = \begin{cases} 1, & \text{if } x_i \in {}^iA_1 \\ 0, & \text{if } x_i \in {}^iA_0 \end{cases} \tag{1.1}$$

Note that ${}^iA_1 \cap {}^iA_0 = \phi$ and ${}^iA_1 \cup {}^iA_0 = R$ (the measure space of X_i). After a decision is made at a sensor, each sensor sends its decision to the FC via a reporting channel that is orthogonal and independent from other sensors' reporting channels. On its turn, the FC makes a global decision as to whether there is a PR or not, using a majority logic decision rule. For the K-out-of-N majority logic decision rule at the FC, we assume N to be an odd number so that ties will not be a possibility. Hence, if the number of ones received is greater than or equal to $K = (\frac{N-1}{2} + 1)$, then the FC determines that the decision $D = 1$ (i.e., 'K' happened), or $D = 0$ (i.e., 'H' happened). The FC global decision rule is given in Equation (1.2).

$$D = \begin{cases} 1, & \text{if } \sum_{i=1}^{N} U_i \geq K \\ 0, & \text{otherwise} \end{cases} \tag{1.2}$$

Note the difference in the definitions between K in Equation (1.2) and the hypothesis symbol 'K'.

1.3 Problem Formulation

To find the set of decision intervals $\{{}^iA_1;\ i = 1,\ ...,\ N\}$ that minimizes the P_e, the derivation starts from the P_e relation and follows the procedure in [40]. By having π_K and π_H as the prior probabilities for both hypotheses 'K' and 'H', respectively, the derivation goes through the following main steps.

$$P_e = \sum_{u_1 \to u_N} P_{0K} + \sum_{u_1 \to u_N} P_{1H} \ , \tag{1.3}$$

where

$\sum_{u_i \to u_j} = \sum_{u_i \in \{0,1\}} \sum_{u_{i+1} \in \{0,1\}} \cdots \sum_{u_j \in \{0,1\}},\ \{i,j\} \in \{1,\ldots,N\},\ i < j,$

$P_{0K} = P(D = 0|U_1^N, K)P(U_1^N, K),$

$P_{1H} = P(D = 1|U_1^N, H)P(U_1^N, H),$

$U_i^j \triangleq \{U_i, U_{i+1}, ..., U_j\}, \forall i, j \in \{1,\ldots,N\}, i < j.$

The main goal in this derivation is to reach an expression of P_e that consists of two types of terms. The first type is the term that does not depend on the value of ${}^1A_1$, while the other type is the term that depends of ${}^1A_1$.

Expressing the relation in (1.3) in terms of $D = 1$, we obtain:

$$P_e = \pi_K \big[\sum_{u_1 \to u_N} P(U_1^N|K) - \sum_{u_1 \to u_N} P_{1|K} \big] + \pi_H \sum_{u_1 \to u_N} P_{1|H}, \tag{1.4}$$

where

$P_{1|K} = P(D = 1|U_1^N)P(U_1^N|K),$

$P_{1|H} = P(D = 1|U_1^N)P(U_1^N|H).$

By simplifying the first group of summations in (1.4) and putting the second and third groups of summations together, the next expression becomes

$$P_e = \pi_K + \sum_{u_1 \to u_N} \big[-\pi_K P_{1|K} + \pi_H P_{1|H} \big]. \tag{1.5}$$

After that, by breaking the first summation (i.e., $\sum_{u_1 \in \{0,1\}}$) into its two terms (i.e., when $u_1 = 0$ and $u_1 = 1$) and using the fact that $P(U_1 = 0, U_2^N|K) = P(U_2^N|K) - P(U_1 = 1, U_2^N|K)$, the relation in (1.5) ends up as follows:

$$
\begin{aligned}
P_e = \pi_K & \\
& + \overbrace{\sum_{u_2 \to u_N} \left[\begin{array}{l} -\pi_K P(D=1|U_1=1,U_2^N,K) \cdot P(U_1=1,U_2^N|K) \\ +\pi_H P(D=1|U_1=1,U_2^N,H) \cdot P(U_1=1,U_2^N|H) \end{array} \right]}^{I_1} \\
& + \overbrace{\sum_{u_2 \to u_N} \left[\begin{array}{l} -\pi_K P(D=1|U_1=0,U_2^N,K) \cdot P(U_2^N|K) \\ +\pi_H P(D=1|U_1=0,U_2^N,H) \cdot P(U_2^N|H) \end{array} \right]}^{II_1} \\
& + \overbrace{\sum_{u_2 \to u_N} \left[\begin{array}{l} \pi_K P(D=1|U_1=0,U_2^N,K) \cdot P(U_1=1,U_2^N|K) \\ -\pi_H P(D=1|U_1=0,U_2^N,H) \cdot P(U_1=1,U_2^N|H) \end{array} \right]}^{III_1}
\end{aligned} \quad . \tag{1.6}
$$

Putting I_1 and III_1 together and simplify the expression by grouping the terms with the common factors, the expression in (1.6) becomes as follows:

$$
P_e = \pi_K + II_1 + \sum_{u_2 \to u_N} (\delta_1 + \beta_1) \quad , \tag{1.7}
$$

where

$$
\begin{aligned}
\delta_1 = \pi_K [& P(D=1|U_1=0,U_2^N,K) \\
& - P(D=1|U_1=1,U_2^N,K)] P(U_1=1,U_2^N|K)
\end{aligned}
$$

$$
\begin{aligned}
\beta_1 = \pi_H [& P(D=1|U_1=1,U_2^N,H) \\
& - P(D=1|U_1=0,U_2^N,H)] P(U_1=1,U_2^N|H)
\end{aligned}
$$

Finally, by representing both $P(U_1=1,U_2^N|K)$ and $P(U_1=1,U_2^N|H)$ in their integral form of the joint pdfs (i.e., $f_K(x_1,x_2,...,x_N)$ and $f_H(x_1,x_2,...,x_N)$ respectively) and by making some simplifications, we get the last expression of P_e that satisfy our goal of having two types of terms according to their dependence on ${}^1A_1$.

$$
P_e = \pi_K + II_1 + \int_{{}^1A_1} \overbrace{\overbrace{\left[\sum_{u_2 \to u_N} P_1 I_1 \right]}^{L_1}}^{M_1} dx_1, \tag{1.8}
$$

where:

$I_1 = \int_{{}^2A_{u_2}} \dots \int_{{}^NA_{u_N}} f(x_1, x_2, ..., x_N)\, dx_2 ... dx_N$

$P_1 = P(D = 1|U_1 = 0, U_2^N) - P(D = 1|U_1 = 1, U_2^N)$

$f(x_1, x_2, ..., x_N) = \pi_K f_K(x_1, x_2, ..., x_N) - \pi_H f_H(x_1, x_2, ..., x_N)$

Note, in the above equation, that the terms named M_1 and L_1 correspond to sensor 1 since we derived the P_e formula with ${}^1A_1$ as a variable.

Now, by fixing all ${}^iA_1, \forall i = 2, ...N$ to some given values, the interval(s) of ${}^1A_1$ can be found in order to minimize the P_e. Note that the first two terms in (1.8) become also fixed by fixing the intervals ${}^iA_1, \forall i = 2, ...N$. Hence, the only term that is left unfixed and can be used to minimize P_e and find ${}^1A_1$ is M_1. Minimizing M_1 means minimizing its integrand L_1. Hence, minimizing L_1 means minimizing P_e and this happens when L_1 is less than zero. According to this, ${}^1A_1$ can be defined as ${}^1A_1 = \{x_1 : L_1 \leq 0\}$. This means if the measured signal for sensor 1 falls in the interval(s) ${}^1A_1$ where L_1 is less than or equal to zero, then the local decision at sensor 1 is hypothesis 'K' and vice versa.

For sensor i, $i = 1, \dots, N$, with the corresponding M_i and L_i terms, the P_e formula has ${}^iA_1$ as a variable and all other intervals ${}^jA_1, j = 1, \dots, N, j \neq i$, are fixed. In a similar way it can be shown that all other sensors have the same definition for their corresponding intervals ${}^iA_1$. Hence, ${}^iA_1 = \{x_i : L_i \leq 0\}$ and ${}^iA_0 = \overline{{}^iA_1},\ i = 1, ...N$.
As a result, by following the GSSA algorithm, all decision intervals $\{{}^iA_1;\ i = 1,\ ...,\ N\}$ that minimize the P_e at the FC can be generated.

1.4 Gauss-Seidel Sampling Algorithm to find the decision intervals

A new algorithm that has two phases is presented in this section. The first phase is the Gauss-Seidel iterative method phase, and the second one is the Sampling Algorithm phase. This new two-phase algorithm is called GSSA (Gauss-Seidel Sampling Algorithm). The pseudo code of this algorithm is given in Algorithm 1. The purpose of the first phase is to let the intervals ${}^iA_1, \forall\ i \in \{1, ..., N\}$ converge to their optima where the minimum value

of P_e happens. This phase starts by fixing the intervals ${}^2A_1$ to ${}^NA_1$ to their initial values. This will make ${}^1A_1$ the only variable interval. Next, GSSA algorithm moves to the second phase (Sampling algorithm phase). This phase is responsible for finding the exact value of ${}^1A_1$ given the nonlinear function L_1(described in Section 1.3). Note that the intervals other than ${}^1A_1$ are known to L_1, since they are fixed but contribute to shaping L_1 as a function of ${}^1A_1$. The Sampling algorithm phase returns back the threshold values that define ${}^1A_1$ in which $L_1 \leq 0$. Next, GSSA returns to phase 1. On its turn, phase 1 makes ${}^1A_1$ fixed to its new value while also fixing all other intervals on their previous values except ${}^2A_1$ which is made the variable for this time. Now, the new value of ${}^2A_1$ can be found by using phase 2 in a manner similar to the one used for ${}^1A_1$ but with the use of L_2. The GSSA algorithm does the same with the rest of intervals and then gets back to ${}^1A_1$ and starts the process again. The GSSA algorithm keeps looping until each interval converges to a specific range (not necessary to be equal for all of them). This convergence is measured by defining an error value (ϵ) which is used to decide if the previous and the current values of any interval are equal or not. Let ${}^iA_{1Old}$ be the old version of ${}^iA_1$, then ${}^iA_1 =^i A_{1Old}$ and ${}^iA_1 \neq^i A_{1Old}$ can be determined as in the following example:
Suppose ${}^iA_{1Old} = (a_{11}, b_{11}) \cup (a_{12}, b_{12})$ and it is represented by a row as: ${}^ir_{Old} = [a_{11}, b_{11}, a_{12}, b_{12}]$. And let ${}^iA_1 = (a_{21}, b_{21}) \cup (a_{22}, b_{22})$ and it is represented by a row as: ${}^ir = [a_{21}, b_{21}, a_{22}, b_{22}]$. Then ${}^iA_1 =^i A_{1Old}$, if the sizes of ${}^ir_{Old}$ and ${}^ir$ are equal and the value $|a_{11} - a_{21}| + |b_{11} - b_{21}| + |a_{12} - a_{22}| + |b_{12} - b_{22}| < \epsilon$. Otherwise, ${}^iA_1 \neq^i A_{1Old}$.

Regarding Sampling Algorithm phase, it is named as it is similar to the sampling concept in signal processing. The use of this phase is to find the thresholds (roots) of the given nonlinear function $L_i, \forall i \in \{1, \ldots, N\}$, and then return the interval(s) of ${}^iA_1$ where this function is non-positive. To simplify the operation of this phase without losing so much accuracy, we can bound our search for thresholds in a specified range. This range can be determined according to the sensor's marginal pdf that corresponds to the given L_i. For

Algorithm 1 GSSA algorithm pseudo code

Phase 1 – Gauss-Seidel iterative method phase

Initialization:
Put initial values for ${}^{2}A_1$ to ${}^{N}A_1$
Put a value for the acceptable error (ϵ)
procedure GAUSS-SEIDEL ITERATIVE METHOD PHASE
    do
        Define ${}^{i}A_{1Old} = {}^{i}A_1, \forall\ i \in \{1, ..., N\}$
        Substitute ${}^{2}A_1$ to ${}^{N}A_1$ in L_1
        go to Phase 2 to find ${}^{1}A_1 = \{x_1 : L_1 \leq 0\}$
        Substitute ${}^{1}A_1,\ {}^{3}A_1$ to ${}^{N}A_1$ in L_2
        go to Phase 2 to find ${}^{2}A_1 = \{x_2 : L_2 \leq 0\}$
        .
        .
        .
        Substitute ${}^{1}A_1$ to ${}^{N-1}A_1$ in L_N
        go to Phase 2 to find ${}^{N}A_1 = \{x_N : L_N \leq 0\}$
    while (${}^{i}A_{1Old} \neq {}^{i}A_1$ for any $i \in \{1, ..., N\}$)
    Return ${}^{i}A_1, \forall\ i \in \{1, ..., N\}$
end procedure

Phase 2 – Sampling algorithm phase

Initialization:
Take in L as a nonlinear function
Define the search range
Put a value for the acceptable tolerance (ζ)
procedure SAMPLING ALGORITHM PHASE
    Divide the range into intervals with width equal to 2ζ
    Examine the sign of L at the center of each interval
    Record an L root at the mid point between two centers having different L signs
    Determine the sign of L between the roots
    Return the interval(s) (A_1) where L is non-positive
end procedure

example, in the simple case of having a single Gaussian RV with mean (μ) 0, we can have the range $(-3\sigma, 3\sigma)$ as our search range, where σ is the standard deviation. This range contains around 97% of the probability. Hence, this property is very helpful in terms of complexity reduction compared to the search in the range $(-\infty, \infty)$. In our work, since we have two marginal pdfs for each sensor, (one under 'K' and the other under 'H'), the defined search range for any RV X_i in the case of multivariate Gaussian will be $([\min(\mu_{K_i}, \mu_{H_i}) - c\max(\sigma_{K_i}, \sigma_{H_i})], [\max(\mu_{K_i}, \mu_{H_i}) + c\max(\sigma_{K_i}, \sigma_{H_i})])$, $i \in \{1, 2, ..., N\}$, where c is a constant that can be specified in advance, such as $c = 3$ in our previous Gaussian RV. In addition, as an assumption after getting the results from GSSA, the intervals $([\max(\mu_{K_i}, \mu_{H_i}) + c\max(\sigma_{K_i}, \sigma_{H_i})], \infty)$ and $(\max(threshold), [\max(\mu_{K_i}, \mu_{H_i}) + c\max(\sigma_{K_i}, \sigma_{H_i})])$ are combined together. And also, the same happens to the intervals $(-\infty, [\min(\mu_{K_i}, \mu_{H_i}) - c\max(\sigma_{K_i}, \sigma_{H_i})])$ and $([\min(\mu_{K_i}, \mu_{H_i}) - c\max(\sigma_{K_i}, \sigma_{H_i})], \min(threshold))$. As can be seen from Algorithm 1, in the Sampling Algorithm phase, the search range is divided into small intervals with the width 2ζ, where ζ represents the maximum tolerance (i.e. acceptable error) in the found root value within any interval.

The advantages of this algorithm compared to other existing root searching algorithms (e.g. [79, 78, 17]) include:

- No use of derivatives.
- Removal of the possibility of dividing by zero.
- No need for initial values to start the algorithm with.
- Guarantee to find all thresholds up to the resolution defined by the tolerance.
- Low complexity and fast operation.

After finding the optimum values of ${}^{i}A_1, \forall i \in \{1, ..., N\}$, that minimizes P_e, we can write the probability of detection (P_D), false alarm (P_F) and P_e as follows:

$$P_D = \sum_{u_1 \to u_N} P_{1|K} \tag{1.9}$$

$$P_F = \sum_{u_1 \to u_N} P_{1|H} \tag{1.10}$$

$$P_e = \pi_K(1 - P_D) + \pi_H P_F \tag{1.11}$$

1.5 Results

We studied the case of multivariate Gaussian density due to its simple closed-form expression and its wide applicability in many signal detection situations.

$N-$RVs $(X_1, X_2, ..., X_N)$ are called jointly Gaussian if their joint pdf can be written as:

$$f(x_1, x_2, ..., x_N) = \frac{|C_x|^{-\frac{1}{2}}}{(2\pi)^{\frac{N}{2}}} \exp\left\{ - \frac{[x - \bar{X}]^T {C_x}^{-1} [x - \bar{X}]}{2} \right\},$$

where

$[x - \bar{X}] = \left[x_1 - \bar{X_1},\ x_2 - \bar{X_2},\ \ldots\ x_N - \bar{X_N}\right]^T$, $C_x = \left[c_{ij}\right],\ \{i, j\} \in \{1, \ldots, N\}$

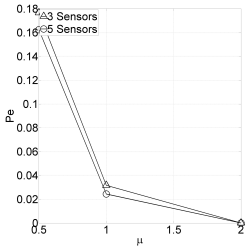
and $[.]^T$, $[.]^{-1}$, $|.|$ stand for matrix transpose, inverse and determinant, respectively. $\bar{X_i} = \alpha_i V$, $i = 1, ...,\ N$, and C_x is the Covariance matrix with

$$c_{ij} = E[(x_i - \bar{X_i})(x_j - \bar{X_j})] = \begin{cases} \sigma^2_{X_i}, & i = j \\ c_{X_i X_j}, & i \neq j \end{cases}$$

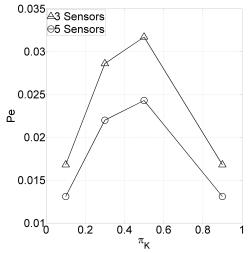
Note that for $i \neq j$, $c_{X_i X_j} = c_{X_j X_i} = \rho_{ij}\sigma_i\sigma_j = \rho_{ji}\sigma_j\sigma_i$. In this work, we used the value of 0.02 for both the error (ϵ) and the tolerance ζ.

First, we consider two arbitrary scenarios as in the following.

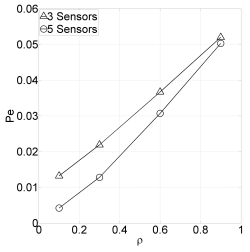
Scenario 1: In $N = 5$ sensors system we have the joint pdf under 'K' is defined by the following matrices with $\pi_K = \frac{2}{3}$, where we assume with different values of the attenuation for each sensor, where $\alpha_1 = 1,\ \alpha_2 = \frac{3}{4},\ \alpha_3 = \frac{11}{12},\ \alpha_4 = \frac{5}{12}$, $\alpha_5 = \frac{6}{12}$, and $V = 12$.



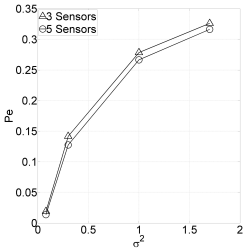
(a) P_e Vs μ for the cases of 3 and 5 sensors.



(b) P_e Vs π_K for the cases of 3 and 5 sensors.



(c) P_e Vs ρ for the cases of 3 and 5 sensors.



(d) P_e Vs σ^2 for the cases of 3 and 5 sensors.

Figure 1.2. P_e curves Vs different parameters change.

$$[x-\bar{X}] = \begin{bmatrix} x_1 - 12 \\ x_2 - 9 \\ x_3 - 11 \\ x_4 - 5 \\ x_5 - 6 \end{bmatrix}, \; C_x = \begin{bmatrix} 0.83 & 0.67 & 0.5 & 0.33 & 0.17 \\ 0.67 & 1.33 & 1 & 0.67 & 0.33 \\ 0.5 & 1 & 1.5 & 1 & 0.5 \\ 0.33 & 0.67 & 1 & 1.33 & 0.67 \\ 0.17 & 0.33 & 0.5 & 0.67 & 0.83 \end{bmatrix}.$$

Where, the joint pdf under 'H' is defined by the following matrices with $\pi_H = \frac{1}{3}$: $[x-\bar{X}] = [x_1\ x_2\ x_3\ x_4\ x_5]^T$, while the covariance matrix under 'H' is 1.2×(the covariance matrix under 'K').

Using GSSA, the test at each sensor for this scenario is defined as follows:

- For sensor 1: ${}^1A_1 = (6.1260, \infty)$
- For sensor 2: ${}^2A_1 = (6.2100, \infty)$
- For sensor 3: ${}^3A_1 = (5.6151, \infty)$
- For sensor 4: ${}^4A_1 = (3.0100, \infty)$
- For sensor 5: ${}^5A_1 = (-2.9940, \infty)$

In this scenario, the probability of error is: $P_e = 1.9 \times 10^{-7}$, the probability of detection is: $P_D \approx 1.0000$ and the probability of false alarm is: $P_F = 4.6472 \times 10^{-7}$.

Scenario 2: It is interesting to present a scenario where one sensor may have multiple non-contiguous intervals instead of only two intervals when making hard decision as in the following example. Assume a system that contains $N = 3$ sensors, with joint pdf under

'K' defined by the following matrices with $\pi_K = 0.5$ while assuming no attenuation on the received signals ($\alpha_i = 1,\ i = 1, 2, 3$) and $V = 1$.

$$[x - \bar{X}] = \begin{bmatrix} x_1 - 1 \\ x_2 - 1 \\ x_3 - 1 \end{bmatrix},\ C_x = \begin{bmatrix} 0.1000 & 0.0900 & 0.0900 \\ 0.0900 & 0.1000 & 0.0900 \\ 0.0900 & 0.0900 & 0.1000 \end{bmatrix}.$$

Let the joint pdf under 'H' be defined by the following matrices with $\pi_H = 0.5$, $[x - \bar{X}] = [x_1\ x_2\ x_3]$, while the covariance matrix under 'H' is the same as the one under 'K'. The GSSA algorithm generates the test at each sensor and the results as follows:

- For sensor 1: ${}^1A_1 = (-0.7887, -0.7487) \cup (-0.6687, -0.5887) \cup (0.4913, 1.4513) \cup (1.5713, 1.8913)$
- For sensor 2: ${}^2A_1 = (0.4913, \infty)$
- For sensor 3: ${}^3A_1 = (0.4913, \infty)$.

In this scenario, the probability of error is: $P_e = 0.0520$, the probability of detection is: $P_D = 0.9534$ and the probability of false alarm is: $P_F = 0.0550$.

Now in another type of simulation, in Figures 1.2(a) to 1.2(d) we assume that the covariance matrices under both 'K' and 'H' are equal while $\sigma_i = \sigma,\ i = 1, ...,\ N$, and $\rho_{ij} = \rho_{ji} = \rho$ for any $i \neq j$, $i = 1, ...,\ N$, $j = 1, ...,\ N$. In Figure 1.2(a), we assume $\rho = 0.5$, $\sigma^2 = 0.1$, $\pi_K = \pi_H = 0.5$, and $\overline{X_i} = \mu,\ i = 1, ..., N$, where μ ranges from 0.5 to 2. This figure shows the P_e curve as a function of μ for both cases $N = 3$ and $N = 5$. As we can see that by increasing μ the value of P_e decreases and this is due to the fact that increasing μ increases the signal to noise ratio (SNR), which decreases the P_e. In Figure 1.2(b), we assume $\rho = 0.5$, $\sigma^2 = 0.1$, π_K changes from 0.1 to 0.9, and $\bar{X}_i = 1,\ i = 1, ..., N$. This figure shows the P_e curve as a function of π_K for both cases $N = 3$ and $N = 5$. As we can see that by increasing π_K the value of P_e increases and reaches its maximum at $\pi_K = 0.5$ then decreasing again. This maximum value happens at $\pi_K = 0.5$ due to the fact that the probability for the received signals to fall in the overlapping volume between the two pdfs under different hypothesis is now maximized. At other values of π_K, the probability to fall in

this area becomes less. In Figure 1.2(c), we assume $\sigma^2 = 0.1$, $\pi_K = 0.5$, ρ changes from 0.1 to 0.9, and $\bar{X}_i = 1,\ i = 1, ..., N$. This figure shows the P_e curve as a function of ρ for both cases $N = 3$ and $N = 5$. It can be seen that the P_e value increases as ρ increases. This is expected because increasing ρ reduces the diversity effect of multiple observations and thus leads to higher P_e values. In Figure 1.2(d), we assume $\pi_K = 0.5$, $\rho = 0.5$, σ^2 changes from 0.08 to 1.7, and $\bar{X}_i = 1,\ i = 1, ..., N$. This figure shows the P_e curve as a function of σ^2 for both cases $N = 3$ and $N = 5$. It can be concluded from this figure that the P_e value increases as σ^2 increases. This is expected because increasing σ^2 means increasing the noise power, which decreases the SNR, and hence, increases the P_e.

It should be noted that in Figures 1.2(a), 1.2(b), 1.2(c) and 1.2(d) the curves related to $N = 5$ has lower P_e values than the corresponding curves for $N = 3$. This is due to the fact that increasing the number of sensors increases the amount of information available at the FC, which makes better decision and gives less P_e value.

We need to admit that using K-out-of-N rule in this paper is a suboptimal fusion. It is worth to mention that the solution generated by GSSA is a suboptimal solution even if we consider all the possible 2^{2^N} fusion rules because changing the initial values of the decision intervals may change the solution generated by GSSA and that may affect the P_e value. In our simulations we conducted three simulation runs with different initial values. In the first two runs, the initial values were different but from our choice, while in the third run the initial values were generated randomly by the computer. The simulations generated almost the same P_e values for the scenarios that we mentioned previously in the three runs, but in some cases we get a totally different decision intervals while having almost the same P_e.

1.6 Conclusion

In this chapter we considered the case of HD based distributed detection with correlated sensing channels. We proposed a generalized problem formulation that is applicable for any kind of joint pdfs with any odd number of sensors. The adopted fusion rule at the

FC is the K-out-of-N rule. This work also proposed a new algorithm called GSSA that can be used to find all the decision rules at all the sensors while having the objective of minimizing the P_e. The results show the usefulness of the derived problem formulation and the effectiveness of the proposed algorithm in obtaining the hard decision rules at all sensors in practical sensor system scenarios.

1.7 Dissertation Organization and Brief Overview

This dissertation target the first two communication layers of wireless networks. The first four chapters are concerned about the spectrum sensing problem at the PHY layer of wireless networks. These chapters propose two new frameworks for spectrum sensing in a correlated environment. These frameworks are hard-decision-based and soft-decision-based, respectively. Bayes criterion for minimizing the probability of error is used to find the distributed decision rule at each sensor in a binary hypothesis multi-sensor system, while considering the correlation over the sensing channels (i.e. among the sensors' measurements).

On the MAC layer, Chapters 5 and 6 present two new channel assignment methods for the coexistence of LTE and WiFi. In these chapters, a new throughput oriented channel assignment mechanism for the infra-structured WiFi networks is introduced to address the randomness problem of its channel assignment. After this, we extend the work to address the new coexistence challenge between LTE and WiFi networks over the unlicensed band.

In this dissertation, the presentation style is that each chapter serves as a stand-alone work. Each chapter includes sections of introduction and problem statement, system model, problem formulation, results, and conclusion.

Finally, Chapter 7 concludes this dissertation as a whole.

CHAPTER 2

Noise Correlation Effect on Detection: Signals in Equicorrelated or Autoregressive(1) Gaussian

In this chapter, we consider the effect of noise correlation on the error performance of binary hypothesis signal detection, when one of two deterministic signals is received in correlated Gaussian noise. For the likelihood ratio detection scheme, analytical performance results are derived for equicorrelated and autoregressive order one models. Although it is known previously that the best signal lies in the direction of eigenvector corresponding to the minimum eigenvalue of the noise covariance matrix, our investigation of the variation of mean signal-to-noise power ratio as a function of correlation parameter (i) shows how correlation leads to increased probability of error up to a point, beyond which monotonic decrease in error probability with increasing correlation is possible and (ii) provides a max-min signal design solution for the unknown correlation parameter case. Numerical results are also included for some specific signals.

2.1 Introduction

A collection of sensors are employed in a variety of situations in order to enhance information gathering and processing operations. Depending upon the capability of sensor networks, either condensed (or pre-processed) information or the raw data from the sensor sites may be combined at a fusion center (FC). The former case is usually referred to as decentralized processing architecture, whereas the latter case is termed as the centralized processing architecture. When signal processing is carried out in order to ascertain whether a target of interest is present in the received signals at the sensor sites or not, the problem becomes a target detection problem.

In this work, we consider the detection of the presence of one of two possible deterministic signals in correlated Gaussian noise. A frequent question that arises is how the statistical correlation among the sensor observations affects the overall detection performance of a detection (centralized as well as decentralized) scheme? Only partial answers to this question seem to be available in the literature. Our study here for a restricted Gaussian observation provides new results for the centralized detection scheme. The literature on the effect of correlation on signal detection is sparse. One reason is that the type of correlation, i.e, the structure of covariance matrix, could be different depending on the models for different scenarios. Second, analytical results are generally possible only for some structured covariance matrices and under asymptotic analysis involving a very large number of sensors, [19], [20]. An analysis of decentralized detection of correlated Gaussian observations involving two sensors, each with one bit quantization, yielded some interesting and surprising results, [76], [25].

In general, it can be expected that the positive correlation among the sensor observations may lead to poor performance when compared to the uncorrelated case, unless the signal vectors under the two hypotheses can be designed appropriately, as can be seen from the textbook example for the two sensor case in [63] and [49], and more recently, for the general case of arbitrary number of sensors [26] . Optimum signal design requires complete knowledge of noise covariance matrix. In some situations, this information may be unavailable or partially known, and we would like to examine, for a given signal set, the variation of probability of detection error (P_e) at the FC, as a function of a correlation parameter. This approach provides a max-min solution to signal design that maximizes the minimum mean signal-to-noise power ratio, as the correlation parameter varies over its admissible interval.

2.2 Problem Formulation

Consider a set of M sensors monitoring a region of interest to ascertain the presence of one of two signals (hypothesis H_1 or H_0) of interest. The signals, which are assumed to

be deterministic, are received in additive Gaussian noise, as shown in (1).

$$\begin{aligned} H_1 &: Z = \mathcal{X}_1 + V \\ H_0 &: Z = \mathcal{X}_0 + V, \end{aligned} \tag{2.1}$$

where $Z^T = [Z_1 \dots Z_M]$, $V^T = [V_1 \ \dots \ V_M]$, $\mathcal{X}_1^T = [\mathcal{X}_{11} \ \dots \ \mathcal{X}_{1M}]$, $\mathcal{X}_0^T = [\mathcal{X}_{01} \ \dots \ \mathcal{X}_{0M}]$. The noise at different sensors are correlated and represented by vector V, which is Gaussian distributed, with zero mean and a positive definite covariance matrix $C = \left[C_{ij}\right]$, $\{i, j\} \in \{1, \dots, M\}$. It is assumed that the signals $\mathcal{X}_1$, $\mathcal{X}_0$ are completely known at the individual sensors and the fusion center. In the centralized test, all sensors send their observations (i.e $Z_j, j = 1, \dots, M$) to the FC through orthogonal and error-free reporting channels. Based on the available information at the FC, the FC makes a final decision on which hypothesis is true using a likelihood ratio test (LRT).

From standard results on LRT for the detection of deterministic signals in correlated Gaussian noise, we decide H_1 if

$$T \geq \gamma \ \equiv \frac{2\log\lambda + \mathcal{X}_1^T C^{-1}\mathcal{X}_1 - \mathcal{X}_0^T C^{-1}\mathcal{X}_0}{2} \tag{2.2}$$

and H_0 otherwise, where $T = Z^T C^{-1}(\mathcal{X}_1 - \mathcal{X}_0)$ and λ equals the ratio of prior probabilities, namely $P(H_0)/P(H_1)$, for Bayes criterion, and equals the value that provides a desired false alarm probability, for Neyman-Pearson criterion. Let $W = C^{-1}(\mathcal{X}_1 - \mathcal{X}_0)$. Probabilities of false alarm and miss detection are given as $P_F = Q\left(\frac{\gamma}{\sqrt{W^T C W}}\right)$ and $P_M = Q\left(\frac{-\gamma + (\mathcal{X}_1 - \mathcal{X}_0)^T W}{\sqrt{W^T C W}}\right)$, respectively. $Q(\cdot)$ is the upper tail function of the standard Gaussian distribution.

2.3 Analysis of Centralized Detection

Define $K = \frac{(E(T|H_1) - E(T|H_0))^2}{Var(T|H_0)}$ as the (mean) signal-to-noise ratio (SNR), also referred to as the deflection coefficient. K can be further simplified as $K = W^T C W$. Without loss of generality, we may assume that under H_0, ${\mathcal{X}_0}^T = [0, \dots, 0]_{1\times M}$ and under H_1, $\mathcal{X}_1$ is replaced

with $\mathcal{X}_1 - \mathcal{X}_0$. For simplicity, let $X = \mathcal{X}_1 - \mathcal{X}_0$ in the sequel. Then, the detection error probability is given as

$$\begin{aligned} P_e &= P(H_1)Q\left(\frac{-\gamma + X^T W}{\sqrt{W^T C W}}\right) + P(H_0)Q\left(\frac{\gamma}{\sqrt{W^T C W}}\right) \\ &= P(H_1)Q\left(\frac{-2\log(\lambda) + K}{2\sqrt{K}}\right) + P(H_0)Q\left(\frac{2\log(\lambda) + K}{2\sqrt{K}}\right) \end{aligned} \tag{2.3}$$

It can be shown that $\frac{dP_e}{dK} < 0$. Similarly, for the NP criterion, for a fixed value of P_F, $\frac{dP_M}{dK} < 0$. Hence, P_e is a monotonic decreasing function with the increase of SNR K, and $P_e \to 0$ as $K \to \infty$.

2.3.1 Equicorrelated Gaussian

In this case, the covariance matrix is specified by $C_{ii} = \sigma^2$, $C_{ij} = \sigma^2\rho$, $i \neq j$. For convenience, σ^2 is taken to be unity in the sequel, as any SNR can be achieved by scaling appropriately the signal level. For this covariance matrix, its inverse, eigenvalues, and eigenvectors are all available in closed-form for finite M, [38]. In fact, the $(i,j)^{\text{th}}$ element of the inverse matrix $D = C^{-1}$ is given by

$$D_{ij} = \begin{cases} \frac{1+(M-2)\rho}{(1-\rho)(1+(M-1)\rho)} & i = j \\ \frac{-\rho}{(1-\rho)(1+(M-1)\rho)}, & i \neq j \end{cases}$$

Our goal is to study the effect of ρ on the detection error probability. By knowing C^{-1}, SNR K can be found as

$$K = b\alpha + (a-b)\beta \tag{2.4}$$

where

$$\alpha = \left(\sum_{j=1}^{M} X_j\right)^2, \ \beta = \sum_{j=1}^{M} X_j^2,$$

$$a = \frac{1+(M-2)\rho}{(1-\rho)(1+(M-1)\rho)} \text{ and } b = \frac{-\rho}{(1-\rho)(1+(M-1)\rho)}.$$

Lemma 1: For any real $X^T = [X_1, \ldots, X_M]$, $0 \leq \frac{\alpha}{\beta} \leq M$.

Proof: The result can be obtained by using Cauchy-Schwartz inequality $|\langle \mathbf{u}, \mathbf{v} \rangle|^{\mathbf{2}} \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle$ and setting $\mathbf{u} = X$ and $\mathbf{v} = \mathbf{1^T}$. □

Because of a monotonic decreasing relationship between P_e and K, $\frac{dK}{d\rho} \geq 0$ is equivalent to $\frac{dP_e}{d\rho} \leq 0$, and vice versa. Hence from (2.4), K increases with ρ if

$$\frac{dK}{d\rho} = \frac{\mathrm{d}b}{\mathrm{d}\rho}\alpha + \left(\frac{\mathrm{d}a}{\mathrm{d}\rho} - \frac{\mathrm{d}b}{\mathrm{d}\rho}\right)\beta > 0. \tag{2.5}$$

Consequently, we can arrive at the following equivalent condition,

$$F(\rho) \triangleq \frac{(1+(M-1)\rho)^2}{1+(M-1)\rho^2} > \frac{\alpha}{\beta}. \tag{2.6}$$

It can be seen that $F(\rho)$ is a monotonic increasing function of ρ and its minimum value of 0 is attained at the minimum value of $\rho = -\frac{1}{M-1}$, whereas its maximum value of M is attained when ρ takes its maximum value of 1. In addition, $F(0) = 1$. Using Lemma 1 and (2.6), we have the following result.

Corrolary 1: The ρ that minimizes SNR K is given as the solution to:

$$F(\rho) \triangleq \frac{(1+(M-1)\rho)^2}{1+(M-1)\rho^2} = \frac{\alpha}{\beta}. \tag{2.7}$$

The above resut shows that the correlation point ρ at which the SNR reaches its minimum value is identical for all signals X that have the same $\frac{\alpha}{\beta}$ value.

Proposition 1: For a good detection performance, it is desirable that $\frac{\mathrm{d}P_e}{\mathrm{d}\rho} < 0$ when $\rho > 0$, and $\frac{\mathrm{d}P_e}{\mathrm{d}\rho} > 0$ when $\rho < 0$.

The proposition simply indicates that the correlation introduced at the sensor sites can be exploited advantageously by a 'good' detection system, thereby providing improved performance beyond that achieved with zero ρ. It is known that, given the knowledge of covariance matrix, the signal X can be chosen as the vector proportional to the eigenvector

corresponding to the minimum eigenvalue of the covariance matrix C so that K is maximized, [63], and [26]. Using the eigen-analysis of the covariance matrix, [38], we have the following lemma.

Lemma 2: For the equicorrelated case, the minimum eigenvalues, corresponding eigenvectors, and SNRs satisfy the following conditions:

For $\rho > 0$, the minimum eigenvalue is $\lambda_{min} = 1 - \rho$ and the corresponding eigenvector ($V_{min} = [V_1, \ldots, V_M]$) satisfies the condition $\sum_{j=1}^{M} V_j = 0$. Notice that the best signal X is proportional to V_{min}. Hence, this requires that $\sum_{j=1}^{M} X_j = 0$.

Setting $\alpha = 0$ in (2.4) we have $K = \frac{\beta}{1-\rho}$ which is the maximum achievable SNR, for a given positive ρ. As $\rho \to 1$, we have $K \to \infty$ and $P_e \to 0$.

For $\rho < 0$, the minimum eigenvalue is $\lambda_{min} = (1 + (M-1)\rho)$ and the corresponding eigenvector V_{min} satisfies the condition $V_1 = V_2 = \cdots = V_M$. Hence, $X = X_1[1 \ \ldots \ 1]^T$.

Putting the corresponding signal in (2.4), we have $K = \frac{\beta}{1+(M-1)\rho}$ which is the maximum achievable SNR, for a given negative ρ. Hence, as $\rho \to -\frac{1}{M-1}$, $K \to \infty$ and $P_e \to 0$. We also see that when $\rho \to 0$, $K \to \beta = M{X_1}^2$.

A few remarks follow.

- (2.5) and (2.6) determine how the SNR (hence the detection error) varies with the variation of correlation for any given signal X. These results are new and have not been seen in the literature.

- What happens if the signal set is chosen as $X = X_1[1 \ \ldots \ 1]^T$ while the correlation happens to be positive? In this case, (2.7) is not satisfied, i.e., $F(\rho) < \frac{\alpha}{\beta} = M$, for any $0 < \rho < 1$, and hence P_e will increase monotonically with respect to ρ. Ultimately, the error rate will be equivalent to that of a single sensor case when $\rho = 1$ (perfect correlation). This result has been known for finite sample and asymptotic cases (e.g.[19], and [26]) although the derivation of (2.5) and (2.6) that leads to this result is new.

- Next, consider the case of a general signal such that $0 < \frac{\alpha}{\beta} < M$. In this case, P_e increases

with ρ until the point ρ^* at which $F(\rho^*) = \frac{\alpha}{\beta}$. When ρ is further increased above ρ^*, P_e decreases and reaches zero when $\rho = 1$ (perfect correlation). Hence, except for the identical signals case, a strong positive correlation is always beneficial. By analogy, we can also conclude that for negative correlation, except for the signal that aligns with the eigenvector corresponding to positive correlation condition, the probability of error will approach zero as ρ approaches $-\frac{1}{M-1}$.

- Max-Min Signal Design

  Without knowing ρ, by utilizing (2.7), we can set up the following max-min problem to find the best signal X^*, among all possible signal sets, that maximizes the minimum SNR achieved, as ρ takes values over the interval $(\frac{-1}{(M-1)}, 1)$:

$$X^* = \arg\max_X(\arg\min_\rho(K)). \tag{2.8}$$

  Solution: Solution to (2.7) provides $\rho^+ = \arg\min_\rho(K)$, with the corresponding K^+ obtained from (2.4) as follows:

$$K^+ = \frac{\beta}{1 + (M-1)(\rho^+)^2}. \tag{2.9}$$

  If we scale each X_i by a constant $\sqrt{g}$, $g > 0$, we end up scaling both K and β by g. Hence, without any loss of generality, we can constrain the power of the signal X as unity, so that $\beta = ||X||^2 = 1$. The value of ρ^+ that is to be used in (2.9) is the solution to (2.7) and hence, it is a function of α, (β fixed at 1). Hence, if there exists an $X^* = X$, which is captured by a particular α, ($\beta = 1$), for which $\rho^+ = 0$ is the solution to (2.7), then K^+ in (2.9) will be maximized. That is, the max-min solution to X in (2.8) is the solution to X in (2.7) with $\rho = 0$. Therefore, the max-min solution X^* satisfies $\frac{\alpha}{\beta} = 1$, which can be

explictly written as follows:

$$\left(\sum_{j=1}^{M} X_j^*\right)^2 = \sum_{j=1}^{M} X_j^{*2} \tag{2.10}$$

- If ρ is fixed at 0, the noise being Gaussian, the noise components at sensors are all independent and hence the optimal signal X could be any signal, with the SNR solely determined by β. However, for the max-min design, although the max-min point corresponds to $\rho = 0$, the optimal max-min signal X^* has to satisfy (2.10).

2.3.2 Autoregressive (AR(1)) Model

In this case, the covariance matrix is given by $C_{ii} = 1, C_{ij} = \rho^{|i-j|}, i \neq j$. This matrix was considered previously in asymptotic analysis involving a large number of sensors [20], [52], and [21]. For this Toeplitz matrix, the inverse is given as follows:

$$C^{-1} = \frac{1}{(1-\rho^2)} \begin{bmatrix} 1 & -\rho & 0 & \dots & 0 \\ -\rho & 1+\rho^2 & -\rho & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -\rho & 1+\rho^2 & -\rho \\ 0 & \dots & 0 & -\rho & 1 \end{bmatrix}.$$

It is easy to show that the SNR (K) in this case is given as follows:

$$K = \frac{-\rho^2}{(1-\rho^2)}\gamma + \frac{-2\rho}{(1-\rho^2)}\eta + \frac{(1+\rho^2)}{(1-\rho^2)}\delta, \tag{2.11}$$

where $\gamma = X_1^2 + X_M^2,\ \eta = \sum_{j=1}^{M-1} X_j X_{j+1}$ and $\delta = \sum_{j=1}^{M} X_j^2$. Hence,

$$\frac{dK}{d\rho} = \frac{-2\rho}{(1-\rho^2)^2}\gamma + \frac{-2(1+\rho^2)}{(1-\rho^2)^2}\eta + \frac{4\rho}{(1-\rho^2)^2}\delta. \tag{2.12}$$

Therefore, $\frac{\mathrm{d}P_e}{\mathrm{d}\rho} < 0$ if

$$G(\rho) \triangleq \frac{\rho}{1+\rho^2} > \frac{\eta}{(2\delta - \gamma)}. \tag{2.13}$$

It can be seen that $G(\rho)$ is a monotonic increasing function of $-1 < \rho < 1$ and that its minimum value of $\frac{-1}{2}$ is approached as ρ approaches -1, whereas its maximum value of $\frac{1}{2}$ is approached when ρ approaches 1. Also, $G(0) = 0$.

Lemma 3: $\frac{\eta}{(2\delta-\gamma)}$ is bounded over the interval $(\frac{-1}{2}, \frac{1}{2})$.

Proof: Using Cauchy-Schwartz inequality: $(\sum_{j=1}^{M-1} a_j b_j)^2 \leq (\sum_{j=1}^{M-1} a_j^2)(\sum_{j=1}^{M-1} b_j^2)$ to the sequences $a_j = X_j, b_j = X_{j+1}$, it can be seen that $\eta^2 \leq (\delta - X_M^2)(\delta - X_1^2)$. The bound follows by taking square root on both sides and then diving by $2\delta - \gamma$. □

Hence, for any given ρ, there exists a signal set that provides the best performance. For this AR(1) model, in general, no closed-form results for the minimum eigenvalue and the corresponding eigenvector are available. For given values of M and ρ, we rely on numerical analysis to design a best signal set and find the corresponding error performance.

Remarks:

- For $M = 3$, explicit solution for eigenvalues and eigenvectors can be obtained easily. One eigenvalue is $1 - \rho^2$. The minimum eigenvalue is given by $\lambda_{min} = 1 - \frac{-\rho^2+|\rho|\sqrt{(\rho^2+8)}}{2}$ and a corresponding eigenvector satisfies $X^T = [x, x_2, x]$, where $x_2 = -x \operatorname{sign} \rho(\frac{(|\rho|+\sqrt{(\rho^2+8)}}{2})$.

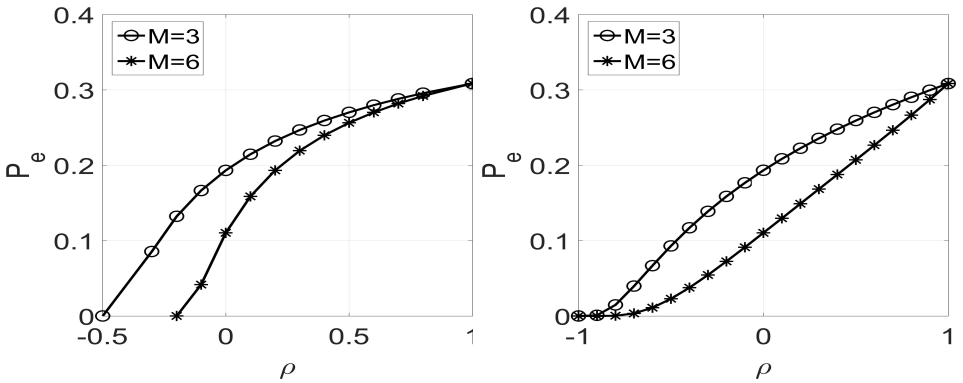
- Max-Min Signal Design

  Proceeding along lines similar to the previous subsection, the max-min solution for the AR(1) model is given by $\rho = 0$ and $\eta = 0$. The corresponding signal X^* satisfies $\sum_{j=1}^{M-1} X_j^* X_{j+1}^* = 0$.

2.4 Numerical Results

The following numerical results are presented for the equicorrelated and AR(1) model cases. In all cases, $\sigma^2 = 1$, $P(H_1) = 0.5$ and two values of $M = 3, M = 6$ are considered. Also, for hypothesis H_0, the signal was assumed to be zero. Then $X = \mathcal{X}_1$. We programmed in $MATLAB^{®}$ to obtain the numerical results.

Scenario 1: Identical signals at all sensors. Signals under H_1 for the two values of M are assumed to be $\mathcal{X}_1 = [1\ 1\ 1]$ and $\mathcal{X}_1 = [1\ 1\ 1\ 1\ 1\ 1]$, respectively. Fig. 2.1(a) and 2.1(b) plot P_e against ρ for this scenario. According to corollary 1 for the equicorrelated case, notice that the signal set assumed is not optimal for ρ positive, but is optimal for negative correlation. In fact, for positive correlation, as ρ tends to 1, the error probability tends to the value achieved with a single sensor.

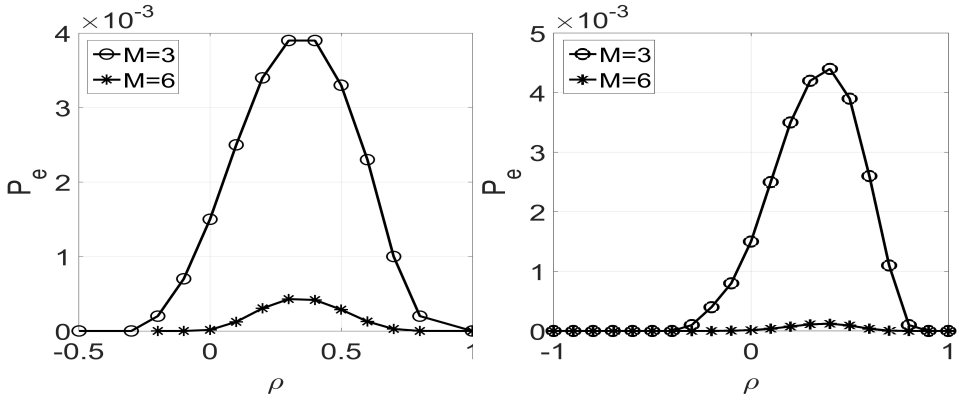


(a) Equicorrelation. (b) Autoregressive.

Figure 2.1. Pe Vs ρ with identical signals at sensors.

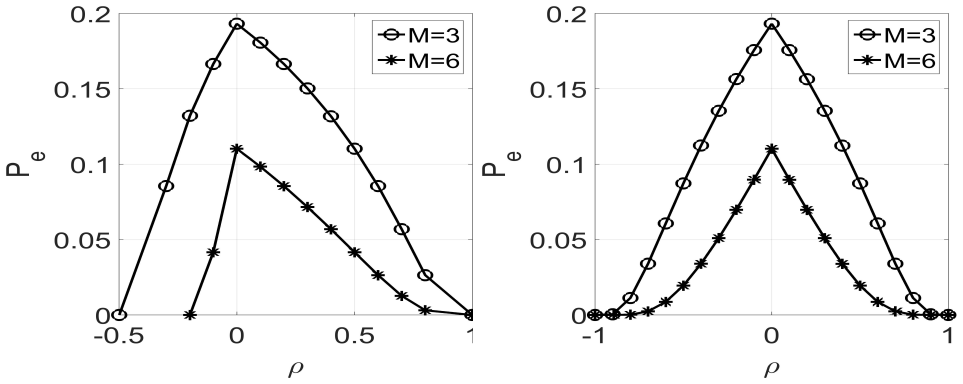
Scenario 2: Signals at sensors are not identical. We assume $\mathcal{X}_1 = [1\ 5\ 3]$ and $\mathcal{X}_1 = [1\ 5\ 3\ 1\ 5\ 3]$. Fig. 2.2(a) and 2.2(b) plot P_e Vs ρ for this scenario. For the equicorrelated case, for ρ greater than a certain positive value, P_e decreases with ρ, thereby providing perfect detection as correlation coefficient approaches 1. A similar behavior is seen for the autoregressive case. As shown analytically, the peak value of P_e is obtained when (7) is satisfied for equicorrelated case (greater relation in (13) becomes equality for AR(1) model).

Scenario 3: Best signal sets. This scenario studies the behavior of P_e with respect to ρ when we have the ability to design the received signals at the sensors using the knowledge of the covariance matrix . Here, we constrain signal power as $||\mathcal{X}_1||^2 = M$, with $\mathcal{X}_1$ as an eigenvector corresponding to the minimum eigenvalue of the covariance matrix. Fig. 2.3(a) and 2.3(b) plot P_e against ρ for the two correlation models.



(a) Equicorrelation. (b) Autoregressive.

Figure 2.2. Pe Vs ρ for not all identical signals.



(a) Equicorrelation. (b) Autoregressive.

Figure 2.3. Pe Vs ρ with best signal design.

In scenarios 1 and 3, the signal power $||\mathcal{X}_1||^2$ equals M, whereas for scenario 2, it is a much higher value (equals 35 for $M = 3$ and 70 for $M = 6$). Hence, the probability of error is significantly lower in scenario 2. For scenario 3, Table 2.1 presents some best signal designs in the case of 3 sensors, for some values of ρ.

ρ	Equicorrelation	Autoregressive
-0.999	-	[0.7071 1.4142 0.7071]
-0.7	-	[0.7549 1.3639 0.7549]
-0.5	[1 1 1]	[0.7871 1.3271 0.7871]
-0.3	[1 1 1]	[0.8191 1.2877 0.8191]
0	[1.7321 0 0]	$[-0.8660\ 1.2248\ -0.8660]$
0.3	$[0.7071\ 0.7071\ -1.4142]$	$[-0.8191\ 1.2877\ -0.8191]$
0.5	$[0.7071\ 0.7071\ -1.4142]$	$[-0.7870\ 1.3271\ -0.7870]$
0.7	$[-1.2633\ 0.0812\ 1.1821]$	$[-0.7548\ 1.3640\ -0.7548]$
0.999	$[1.1953\ 0.0569\ -1.2522]$	$[0.7071\ -1.4142\ 0.7071]$

Table 2.1. Best signals ($\mathcal{X}_1^T$) designed for $M = 3$ in scenario 3.

From Table 2.1, for equicorrelation, we see that $\sum_{j=1}^{M} X_j$ is M for negative ρ and is 0 for positive ρ, thus agreeing with Lemma 2. For positive correlation, (M-1) eigenvalues are repeated at minimum values, thus the eigenvector corresponding to the minimum eigenvalue lies in the spanned (M-1) dimensional space. For negative correlation (recall $\rho > \frac{-1}{M-1}$), there is a single, non-repeated minimum eigenvalue. For the autoregressive model, results in Table 2.1 agree with the first remark under Lemma 3.

2.5 Conclusion

In this chapter, for the centralized detection of one of two deterministic signals in Gaussian noise, with either equicorrelated covariance or AR(1) model covariance, we show analytically how the correlation between the sensors affects the detection performance for any signal set. We also derived a max-min solution for the signal set that achieves the maximum of minimum SNR, as ρ takes values over the admissible range. Numerical results are presented to show the behavior of P_e as a function of ρ, for representative signal sets.

CHAPTER 3

Soft-Decision Based Distributed Detection With Correlated Sensing Channels

Prior works on distributed detection with correlated Gaussian observations primarily dealt with single-bit reports to the fusion center and suboptimal fusion rules. Considering multiple bits and independent reporting channel errors, we formulate the optimal detection problem as a nonlinear integer programming problem for which the genetic algorithm is applied to find a suboptimal solution. The Bayesian probability of error performance of our suboptimal solution, first, for one-bit, is better than the performance of some previously proposed schemes and, second, for multiple bits, approaches that of the centralized detection scheme.

3.1 Introduction

3.1.1 Motivation

Being an indispensable part of any cognitive radio system, spectrum sensing (SS) gains huge attention in the literature (see [21] and the references therein). When multiple sensors could cooperate and the bandwidth of the reporting channels between sensors and their fusion center (FC) (i.e., the network entity that generates the global network decision) is limited, distributed detection (DD) systems are popularly adopted for sensing, [44, 76, 22, 18, 50]. A DD system provides the ability for the sensors to quantize (compress) the detected (received) signal (information) into a single-bit local decision (hard-decision) or a multibit local decision (soft-decision), [57, 44, 76, 22, 18]. However, detection in the case of correlated observations at the sensors in a DD system is still not fully understood, [44, 76, 71]. For example, in [76], general rules that govern the optimal sensors' decisions given the FC rule in a two-sensor system were only obtained in some limited scenarios, for the bivariate Gaussian model.

Many works in the literature (e.g. [22] and [18]) study both hard and soft decision based DD systems without considering the dependency between the sensors' readings. Such systems perform poorly in terms of the probability of error, P_e, at the FC (a.k.a: fusion-error probability) if correlation exists among the sensors' readings. This can lead to poor receiver operating curve as well, if a NeymanPearson framework is considered. The works in [57, 44, 76], and [50] study the case of hard-decision in a correlated sensing environment using a person-by-person (PBP) technique to find sub-optimal quantization rules at the sensors while minimizing the P_e at the FC. These works have not considered the soft-decision rules at the sensors even if the reporting channel bandwidth allows multibit local decisions reported from the sensors to the FC. Another issue in these works is the use of some special cases of the counting rule ([27] and [29]), such as AND, OR, and XOR, in [57] and [76], and the majority rule, in [44] and [50], as fusion rules at the FC. None of these fusion rules is proven to be always optimal in an environment with correlated readings. In order to improve the spectrum detection performance in cognitive radio systems, this paper presents a new generalized framework for DD SS systems.

3.1.2 Contribution and Comparison

The main contribution of this paper is to formulate the problem as an integer non-linear programming (INLP) problem for which a Genetic Algorithm (GA) is applied to find suboptimal solutions for various scenarios. The proposed framework possesses the following important features.

- Considers the case of correlated sensors' readings and the ability to handle both hard and soft-decision rules at the sensors, unlike the works presented in [44], [76], [50], and [57] that handle only the hard decisions when considering correlation.
- Considering the bandwidth constraint on the reporting channels, which limits the number of reporting bits, unlike the work proposed in [26], [20], and [13].
- The ability to handle multiple sensors while considering the correlation with any type

of joint probability density function (pdf) between their readings. Previous works such as [57, 76], and [36] consider two sensors only. Some other works, such as [71], consider multiple sensors but assume identically distributed readings.

- The use of the likelihood ratio test (LRT) to generate the FC global decision, unlike the works in [57, 44, 76], and [50] that consider the AND, OR, XOR, or the majority logic as their fusion rules. Our method updates both local decisions (i.e., thresholds, and codewords assignment) and the fusion rule, simultaneously.

- The ability to handle DD systems with erroneous reporting channels and correlated sensors' readings. In [32], for the one-bit sensor quantization, the authors capture the joint distribution of decisions of sensors using a set of correlation coefficients. In the special case, where the joint distribution is indexed by a single correlation parameter, the performance of a LRT at the FC is examined for several cases of sensor probability of false alarm and probability of detection values. Kam *et al.* [32] consider Bayes error formulation for correlated local decisions. All the previously mentioned DD works assume correlated sensors' readings and error-free reporting channels.

- The use of the GA algorithm in [41] to find a suboptimal decision rules at the sensors and the optimal FC fusion rule (using the LRT), simultaneously. GA has shown potential to get out of local optimum points. In contrast, the PBP algorithm (used in [57, 44, 76], and [40]) has a disadvantage of being likely trapped at a local optimum [21].

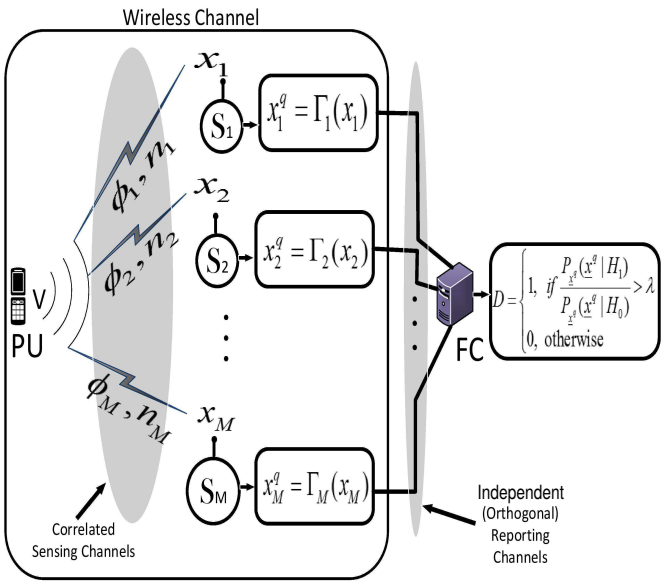


Figure 3.1. Soft decision based sensing system.

3.1.3 Related Work

In [76], DD system with binary hypotheses was considered. The authors interested in the two-sensor case with correlated readings. Three fusion rules were studied, namely, the AND, the OR, and the XOR in terms of the P_e performance at the FC. The bivariate Gaussian joint pdf under both hypotheses was assumed. The decision rules at the sensors were found by using the Gauss-Seidel iterative method (a PBP-based method). Chapter 1 of this dissertation and [44] studied the hard-decision case in correlated readings environment. The PBP technique was used to find sub-optimal decision rules at the sensors to achieve the objective of minimizing the P_e at the FC. The FC employed the majority logic rule to generate the global decision.

In [71], the authors considered the case of correlated readings at the sensors in a DD system. Their study was built on the assumption of identically distributed readings at the sensors in the detection system. In [24], the authors proposed a new framework for DD systems with conditionally dependent observations. The proposal is to add a new random variable (called hidden random variable) between the hypotheses and the observations such that the observations become independent conditioned on this random variable and the hy-

pothesis in force. Under this framework, the authors tested two classical scenarios: detection of deterministic signals in dependent noise and the detection of random signals in an independent noise. Even though this paper provides the possibility to reduce the complexity of the problem once the hidden variable is known, it is still still unclear how to completely describe this hidden random variable. In [58], the authors studied the capacity effect of the reporting channels, between the sensors and the FC, on the asymptotic error exponent (i.e., the error exponent when the number of sensors is very large). This paper considered DD system with both dependent and independent sensor observations. The method presented in [24] (i.e., the hierarchical conditional independence method) was used to test the validity of the results.

In [26], the authors proposed a new framework to select the sparsest possible sub-set of sensors from the whole set of sensors to generate the global decision at the FC. The selection was done such that a desired P_e (or a simpler cost e.g. distance between the conditional distributions of the sensor observations) is met. It is worth mentioning that the selected sensors send their exact readings to the FC without compression. Both cases of correlated and uncorrelated readings are considered in this paper with the a focus on the uncorrelated case.

In Chapter 2 and [46], the effect of correlation on the P_e performance at the FC in a centralized detection system was studied. The focus was on the binary testing with Gaussian distributed readings under two hypotheses with the same covariance matrix but different means. Analytical results were provided using two covariance models: the equicorrelated and the autoregressive models. The results show that the correlation can increase the P_e at the FC up to a limit after which the P_e starts to decrease. For the case when the correlation is unknown, a max-min signal design was provided to achieve the best possible P_e at the FC. This paper shows that the correlation could be exploited for signal detection in the centralized case.

One-bit quantization at the sensors and corresponding fusion center rules have been

studied under the situation of incomplete knowledge of the noise density in the sensor's observation. For example, knowing the noise density to be only symmetric and unimodal, the authors in [28, 30] discuss the performances of a Generalized Likelihood Ratio Test (GLRT), Rao test, and a Generalized Rao test. The Rao test is an asymptotic optimal test under the weak signal condition, but has lower computational complexity when compared to a GLRT. Other results involving one-bit quantization at the sensors have also been reported in the literature [27, 29]. For example, in [27], the authors discuss under the assumption of unknown detection probabilities of the individual sensors decisions, the performances of counting rules (CR) which includes Boolean AND and OR rules as special cases, and locally optimal rule at the FC, which is optimal for weak signal condition. Also, in [29] the authors discussed a general framework of independent Bernoulli trials in which the success probabilities could change with trial index. It was shown in this case that the CR is a uniformly most powerful invariant (UMPI) test, under the invariance group defined by the set of permutations of sensors decisions. However, in both cases, [27] [29], the sensor observations are assumed conditionally independent, given the hypothesis, and hence these models do not consider correlated observations at the sensors.

Copula based multidimensional probability distribution was considered in applications to heterogeneous sensor data with multiple sensors [42]. However, for a specific application, the choice of a particular copula model among several has to be determined by trial and error. Moreover, it will also involve the estimation of parameters associated with a particular model. In principle, given a known copula model that fits a particular detection problem, it is possible to apply the procedure outlined in this paper and obtain a quasi-optimal quantization scheme coupled with the LRT at the FC. However, an investigation of this is beyond the scope of this chapter. In this chapter, we assumed the complete knowledge of the joint pdf that includes the effect of correlation in sensor readings.

Table 3.1 summarizes the main symbols to be used.

Symbol	Definition
M	The total number of sensors
X_j	The received signal random variable at Sensor j
X_j^q	The quantized signal at Sensor j
X_j^r	The received signal through the j^{th} erroneous reporting channel
$\Gamma_j(x_j)$	Decision (quantization) rule at sensor j
D	FC global decision
ϕ_j	The attenuation coefficient on the j^{th} sensing channel
n_j	Noise component on the j^{th} sensing channel
Q_j	Number of quantization levels at Sensor j
N_i	Number of sub-intervals on Sensor $j's$ sensing range
$\epsilon\%$	Mutation percentage in GA
$\alpha_{i_j}^{(j)}$	Quantization level value in sub-interval i_j of Sensor j
P_{b_j}	Probability of bit error on the j^{th} reporting channel
$\mathcal{G}$	The set of solutions in the GA
$Iter$	The total number of GA iterations
w_j	The binary representation of $(X_j^q - 1)$

Table 3.1. Symbols summary.

3.2 Network model and Problem Formulation

3.2.1 An example of Using Correlation

It is often accepted that correlation impairs the detection performance at the FC. It is also shown in Chapter 2 and in [46] that in the centralized case, strong correlation could be helpful for a large class of signal sets. To motivate the need of designing a DD system with the consideration of correlated observations, we provide in this section a simple example and illustrate the effect of correlation on system performance. Following this, we provide problem formulation for the cases of both error-free and erroneous reporting channels.

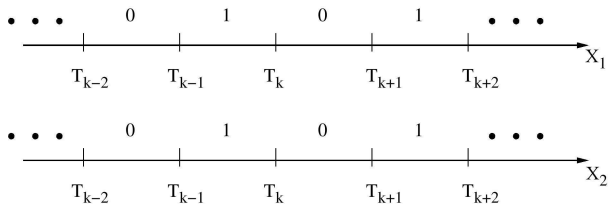


Figure 3.2. Local decision rule at the two sensors

3.2.2 An example of Using Correlation

Let (X_1, X_2) be the observations of two sensors, where (X_1, X_2) are bivariate Gaussian distributed such that under H_0, $(X_1, X_2) \sim \mathcal{N}(0, 0, \sigma_1, \sigma_2, \rho)$ and under H_1, $(X_1, X_2) \sim \mathcal{N}(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$. With one-bit hard decision at each sensor, as $\rho \to 1$, we have the following result.

Theorem 1. *When $\rho = 1$, for two sensors with $\mu_1 \neq \mu_2$, one-bit decision from each sensor can decide the true hypothesis with detection error $P_e = 0$.*

Proof. When $\rho = 1$, observations in both sensors experience the same noise. Under H_0, $X_1 = X_2 = Z$, where Z is the noise, while under H_1, $X_1 = \mu_1 + Z$ and $X_2 = \mu_2 + Z$, where μ_i is the mean value of $X_i, \forall i = 1, 2$. Therefore, testing H_0 and H_1 becomes deciding whether the observations from the two sensors are the same. This can be accomplished by transmitting only one bit from each sensor to the FC, with the local decision rule shown in Fig. 3.2. There are an infinite number of thresholds T_k, where k belongs to the set of integer numbers. Furthermore, $T_{k+1} - T_k = |\mu_1 - \mu_2|$ for all k. Decision 0 and 1 are alternatively assigned to the intervals. Clearly, if $\mu_1 \neq \mu_2$, the FC can perfectly decide H_1 or H_0 using the "XOR" rule. □

Although the Gaussian distribution is considered through this chapter, this theorem also holds for any joint distribution as long as the mean values are different.

This example shows that the perfect detection is possible with just one bit quantization at each sensor, when the correlation coefficient equals 1. This can be taken as the counterpart to the centralized case that we found in [46]. In fact, it also implies that when $\rho = 1$, no more than two sensors are needed for perfect detection. Although this example

shows a limiting case, it is still surprising that perfect detection could be achieved with one bit local decision, which strongly suggests the need to take the correlation into account in a DD system.

3.2.3 Error-free Reporting Channels

Our framework adopts the system model shown in Fig. 3.1. Assume there are M sensors randomly deployed in a certain area. Each Sensor j, i.e., S_j, $j = 1, \ldots, M$, has Q_j levels of quantization. Upon sensing the reading x_j, S_j quantizes it into $x_j^q \in \{1, 2, \ldots, Q_j\}$, where the letter "$q$" associated with any symbol in this chapter refers to the word "*quantized*". To find x_j^q, S_j uses the quantization rule $\Gamma_j(x_j)$. Then it sends x_j^q through an independent error-free reporting channel to the FC.

This system is a binary hypothesis testing system that concerns about the existence of a signal on a given frequency channel (hypothesis H_1) or the absence of the signal (hypothesis H_0). This is useful in cognitive radio networks [74] (and later on in our LTE-WiFi coexistence system model proposed in Chapter 6), where spectrum sensing is needed to find the spectrum holes [61] and make the coexistence on a given frequency band between heterogeneous networks possible. Assume that the signal level transmitted by a primary radio user (PU) in cognitive radio networks under H_1 is V and under H_0 is zero. Then x_j can be modeled under H_1 as $x_j = \phi_j V + n_j$, where ϕ_j and n_j are the attenuation and noise components of sensing channel j between the signal source and S_j, respectively. And under H_0, x_j can be given as $x_j = n_j$. Here, it is assumed that all the noise components $n_j, \forall j = 1, \ldots, M$ are correlated under both H_1 and H_0. Assume that $\underline{x}^q = [x_1^q \ \ldots x_M^q]$, the vector of quantization levels received at the FC from all the sensors, then with $\underline{x}^q$ at the FC, it generates the global decision $D \in \{0, 1\}$ about whether H_0 (i.e., $D = 0$) or H_1 (i.e., $D = 1$) is true using the LRT. It is worth mentioning that in centralized detection the sensors send the exact $x_j, \forall j = \{1, \ldots, M\}$, values to the FC, and then the FC uses the LRT to generate the global decision, [63]. This method provides the best possible P_e performance but, on the

other hand, it is bandwidth inefficient.

In the DD SS system presented in this chapter, we first divide the sensing range of Sensor j (i.e., form $-\infty$ to ∞) into N_j subintervals. This can be achieved by dividing the sensing range j into intervals with equal probability mass $\zeta_j = \frac{1}{N_j}$, under the marginal pdfs of S_j (i.e., $P(H_1)f_{X_j|H_1}(x_j) + P(H_0)f_{X_j|H_0}(x_j)$), where $P(H_1)$ and $P(H_0)$ are the prior probabilities of H_1 and H_0, respectively. Typically, we choose N_j to be larger than Q_j. For S_j, assume the subinterval i is represented by the symbol $I_i^{(j)}, \forall i = 1, \ldots, N_j; j = 1, \ldots, M$. Let the set of variables that correspond to the subintervals of S_j be $\{\alpha_1^{(j)}, \ldots, \alpha_{N_j}^{(j)}\}$, where $\alpha_i^{(j)} \in \{1, \ldots, Q_j\}, \forall i = 1, \ldots, N_j; j = 1, \ldots, M$. For example, if $\alpha_3^{(2)} = 5$, it means when the received signal by S_2 (i.e., x_2) falls in sub-interval 3, then it is quantized to $x_2^q = 5$. Consequently, the probability of error is

$$\begin{aligned} P_e = &\sum_{\underline{x}^q} P(D=0|\underline{x}^q, H_1) P_{\underline{X}^q}(\underline{x}^q|H_1) P(H_1) \\ &+ \sum_{\underline{x}^q} P(D=1|\underline{x}^q, H_0) P_{\underline{X}^q}(\underline{x}^q|H_0) P(H_0) \end{aligned}, \tag{3.1}$$

$$\text{where } \sum_{\underline{x}^q} = \sum_{x_1^q=1}^{Q_1} \cdots \sum_{x_M^q=1}^{Q_M}, \ \underline{x}^q = \{x_1^q, x_2^q, \ldots, x_M^q\},$$

$$\begin{aligned} P(D=1|\underline{x}^q, H_l) &= P(D=1|\underline{x}^q) = 1 - P(D=0|\underline{x}^q) \\ &= \begin{cases} 1, & \text{if } \frac{P_{\underline{X}^q}(\underline{x}^q|H_1)}{P_{\underline{X}^q}(\underline{x}^q|H_0)} \geq \lambda \\ 0, & \text{otherwise} \end{cases} \quad l = 0, 1 \end{aligned} \tag{3.2}$$

where $\lambda = \frac{P(H_0)}{P(H_1)}$. Noticing that $\alpha_i^{(j)}$ can have only one value from the set $\{1, \ldots, Q_j\}$ given the quantization rule $\Gamma_j(x_j)$, these values are mutually exclusive for each individual subinterval. The intersections between the subintervals in the $x_1, x_2, \ldots, x_M$ space are called the subspaces. The sum of all subspaces' probabilities under H_1 that are represented by a specified combination of the $x_1^q, x_2^q, \ldots, x_M^q$ values is given by $P_{\underline{X}^q}(\underline{x}^q|H_1)$. Using these facts,

we can express the term $P_{\underline{X}^q}(\underline{x}^q|H_1)$ in the following form.

$$P_{\underline{X}^q}(\underline{x}^q|H_1) = \sum_{\underline{i}} \prod P_{\underline{I}|H_1}, \tag{3.3}$$

$$\text{where } \sum_{\underline{i}} = \sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \cdots \sum_{i_M=1}^{N_M},$$

$$\prod = \psi(x_1^q - \alpha_{i_1}^{(1)})\psi(x_2^q - \alpha_{i_2}^{(2)}) \ldots \psi(x_M^q - \alpha_{i_M}^{(M)}),$$

$$\psi(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$$

and $\underline{I}$ is the subspace that results form the intersection of the sub-intervals set $\{I_{i_1}^{(1)}, I_{i_2}^{(2)}, \ldots, I_{i_M}^{(M)}\}$. Equation (3.3) is used to find the total probability of the subspaces with the same quantization combination, under H_1. Assume that the thresholds of the subintervals of S_j on axis j are given by the vector $E_j = [e_1^{(j)} = -\infty, e_2^{(j)}, e_3^{(j)}, \ldots, e_{N_j+1}^{(j)} = \infty], \forall j = 1, \ldots M$. Then the E_j elements' values can be determined according to the equal-probability division procedure described previously. Hence

$$P_{\underline{I}|H_1} = P(\underline{I}|H_1) = \int_{e_{i_1}^{(1)}}^{e_{i_1+1}^{(1)}} \ldots \int_{e_{i_M}^{(M)}}^{e_{i_M+1}^{(M)}} f_{\underline{X}|H_1}(\underline{x}) \underline{dx},$$

where, $f_{\underline{X}|H_1}(\underline{x}) = f_{X_1,\ldots,X_M}(x_1, \ldots, x_M)$ is the joint pdf of the random variables $X_1, \ldots, X_M$ under H_1 and $\underline{dx} = dx_1 \ldots dx_M$.

In a similar way, we have:

$$P_{\underline{X}^q}(\underline{x}^q|H_0) = \sum_{\underline{i}} \prod P_{\underline{I}|H_0}. \tag{3.4}$$

Similarly, the product $P(D=0|\underline{x}^q)P_{\underline{X}^q}(\underline{x}^q|H_1)$ can be given as

$$P(D=0|\underline{x}^q)P_{\underline{X}^q}(\underline{x}^q|H_1) = (1-P(D=1|\underline{x}^q)) \sum_{\underline{i}} \prod P_{\underline{I}|H_1} \tag{3.5}$$

We also have

$$P(D=1|\underline{x}^q)P_{\underline{X}^q}(\underline{x}^q|H_0) = P(D=1|\underline{x}^q) \sum_{\underline{i}} \prod P_{\underline{I}|H_0} \tag{3.6}$$

As a result, the optimization problem to be solved is to find the values of $\{\alpha_1^{(j)}, \ldots, \alpha_{N_j}^{(j)}\}, \forall j = \{1, \ldots, N_j\}, i = \{1, \ldots, M\}$ (i.e., $\Gamma(x_j), \forall j$) that minimize the P_e at the FC. Combining (3.1)-(3.6) and with some simple algebraic manipulations, we can formulate the optimization problem as follows

$$\min_{\alpha_{i_j}^{(j)}} \Big(P_e = P(H_1) + \sum_{\underline{x}^q} \sum_{\underline{i}} \delta \prod P(D=1|\underline{x}^q) \Big) \tag{3.7}$$

s.t.

$$P(D=1|\underline{x}^q) = \begin{cases} 1, & \text{if } \sum_{\underline{i}} (P_{\underline{I}|H_1} - \lambda P_{\underline{I}|H_0}) \prod \geq 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\alpha_{i_j}^{(j)} \in \{1, \ldots, Q_j\}, \forall i_j = 1, \ldots, N_j; j = 1, \ldots, M,$$

where $\delta = P_{\underline{I}|H_0}P(H_0) - P_{\underline{I}|H_1}P(H_1)$,

$$\prod = \psi(x_1^q - \alpha_{i_1}^{(1)})\psi(x_2^q - \alpha_{i_2}^{(2)}) \ldots \psi(x_M^q - \alpha_{i_M}^{(M)}),$$

$$\psi(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\sum_{\underline{x}^q} = \sum_{x_1^q=1}^{Q_1} \cdots \sum_{x_M^q=1}^{Q_M} \text{ and } \sum_{\underline{i}} = \sum_{i_1=1}^{N_1} \cdots \sum_{i_M=1}^{N_M}.$$

Note that the total number of variables (T_N) in this optimization problem is $T_N = \sum_{j=1}^{M} N_j$. This problem formulation is one contribution of this work, since it provides the ability to handle multi-level quantization at the sensors and optimal global decision rule at the FC, simultaneously, while assuming the correlation between the sensor readings with the objective of minimizing the P_e at the FC. This optimization problem is an INLP problem, which is considered as an NP-Hard problem. One of the well-known algorithms to solve such kind of optimization problems is the GA. In Section 3.3, the GA is modified and applied to solve this optimization problem.

3.2.4 Erroneous Reporting Channels

We also consider the case of having probability of bit error on each of the reporting channels. All the reporting channels are assumed to be independent binary symmetric channels. The system model used here is the same as the one explained in the previous subsection (i.e., Fig. 3.1) but with erroneous reporting channels. Assume the probability of bit error on reporting channel j is $P_{bj},\ \forall j = 1, \ldots, M$. In this system model, the FC receives random variable X_j^r on reporting channel $j,\ \forall j = 1, \ldots, M$, where X_j^r is the received quantization level by the FC after Sensor j sends the bit-sequence of $(X_j^q - 1)$ (note the "-1" is used because the least value of X_j^q is 1 not 0) through the reporting channel j (note the superscript r refers to the word "*received*"). The received bit-sequence of $(X_j^r - 1)$ might not be the same as the sent bit-sequence of $(X_j^q - 1)$ because of the bit error probability (i.e., P_{bj}) on the reporting channel j. It is also assumed that all $P_{bj},\ \forall j = 1, \ldots, M$ are independent and the P_{bj} affects each bit in the bit-sequence of X_j^q independently from the other bits in the

sequence. The optimal fusion rule at the FC in this system model is the LRT [same as the one in Equation (3.2) with replacing q by r]. Note that $\underline{X}^r = [X_1^r, X_2^r, \ldots, X_M^r]$ (i.e., the random variables vector received at the FC), $\underline{x}^r = [x_1^r, x_2^r, \ldots, x_M^r]$ (i.e., the realizations of $\underline{X}^r$), and $P_{\underline{X}^r}(\underline{x}^r|H_j)$ is the probability that $\underline{x}^r$ is received at the FC under $H_j, \forall j = 0, 1$.

$P_{\underline{X}^r}(\underline{x}^r|H_j)$ can be given as follows

$$\begin{aligned} P_{\underline{X}^r}(\underline{x}^r|H_j) &= \sum_{\underline{x}^q} P_b P_{\underline{X}^q}(\underline{x}^q|H_j) \\ &= \sum_{\underline{x}^q} P_b \sum_{\underline{i}} \prod P_{\underline{I}|H_j} \\ &= \sum_{\underline{x}^q} \sum_{\underline{i}} P_b \prod P_{\underline{I}|H_j}, \forall j = 0, 1 \end{aligned} \tag{3.8}$$

where

$$P_b = P_{b1}^{h_1}(1 - P_{b1})^{b_1 - h_1} \ldots P_{bM}^{h_M}(1 - P_{bM})^{b_M - h_M},$$

and h_j is the Hamming distance between the bit-stream of $(x_j^q - 1)$ and the bit-stream of $(x_j^r - 1)$, $\forall j = 1, \ldots, M$, and b_j is the number of bits in the bit-stream of $(x_j^q - 1)$ (or $(x_j^r - 1)$), $\forall j = 1, \ldots, M$.

By following similar steps to the error-free case, the optimization problem for the system model with reporting errors can be written as follows

$$\min_{\alpha_{i_j}^{(j)}} \left(P_e = P(H_1) + \sum_{\underline{x}^r} \sum_{\underline{x}^q} \sum_{\underline{i}} P_b \delta \prod P(D = 1|\underline{x}^r) \right) \tag{3.9}$$

s.t.

$$P(D = 1|\underline{x}^r) = \begin{cases} 1, & \text{if } \sum_{\underline{x}^q} \sum_{\underline{i}} P_b \prod (P_{\underline{I}|H_1} - \lambda P_{\underline{I}|H_0}) \geq 0 \\ 0, & \text{otherwise} \end{cases}.$$

$$\alpha_{i_j}^{(j)} \in \{1, \ldots, Q_j\}, \forall i_j = 1, \ldots, N_j; j = 1, \ldots, M,$$

where $\sum_{\underline{x}^r} = \sum_{x_1^r=1}^{Q_1} \cdots \sum_{x_M^r=1}^{Q_M}$.

3.3 Genetic Algorithm

In this section, we first present the proposed GA-based decentralized detection algorithm and then discuss its complexity in terms of computational time and memory requirement.

3.3.1 GA procedure

The adopted GA algorithm has the following steps.

Step 1: Generate a set $\mathcal{G}$ of solutions (i.e., a generation/set of chromosomes). Let the number of solutions in this set be $|\mathcal{G}|$ and generate these solutions by randomly assigning values to the "α"s in (3.7) for error-free reports, or (3.9) for erroneous reports.

Step 2: Encode each α value in each solution of $\mathcal{G}$ by representing it into binary sequence.

Step 3: Evaluate the fitness function for each solution in $\mathcal{G}$ (i.e., $1 - P_e$ of each solution).

Step 4: Take a number of solutions that have the highest fitness function values from $\mathcal{G}$ and directly place them into the next generation of $\mathcal{G}$. In this work, the chosen number of solutions is $\frac{|\mathcal{G}|}{2}$.

Step 5: Choose $\frac{|\mathcal{G}|}{4}$ pairs from the current $\mathcal{G}$ according to their fitness value by using the roulette wheel selection method. Then, perform c number of random crossovers between each pair to generate two new solutions. Doing this for all $\frac{|\mathcal{G}|}{4}$ chosen pairs produces $\frac{|\mathcal{G}|}{2}$ new solutions. These new solutions are used to fill the second half of the next generation (i.e., the $\mathcal{G}$ of the next iteration).

Step 6: Perform mutation on the binary version of the new solutions resulted from step 5 by flipping each bit with probability $\frac{\epsilon}{100}$.

Step 7: Repeat steps 2 to 6 for $Iter$ number of iterations.

Step 8: Take the highest fitness value solution from the resulting $\mathcal{G}$ as the final solution of (3.7) in the error-free reports case, or (3.9) for erroneous reports case.

3.3.2 GA Complexity

3.3.2.1 GA Time Complexity

There are two aspects to measure the time complexity of the GA algorithm.

1) *Number of fitness function calculations:* The total number of fitness function calculations in Step 3 can be easily found as $|\mathcal{G}| + (Iter - 1) \times \frac{|\mathcal{G}|}{2} \simeq \frac{Iter \times |\mathcal{G}|}{2}$.

2) *Fitness function calculation time complexity:* To evaluate the fitness of one solution we need to look at the problem formulation.

In the error-free reports case: In this case we have the following.

- Calculating $P(D = 1|\underline{x}^q)$ can be approximated by calculating "$(P_{\underline{I}|H_1} - \lambda P_{\underline{I}|H_0}) \prod$" for "$\prod_{j=1}^{M} N_j$" times, where $\prod_{j=1}^{M} N_j$ is the total number of $\underline{i}$ combinations in $\sum_{\underline{i}}$. Note that calculating $(P_{\underline{I}|H_1} - \lambda P_{\underline{I}|H_0}) \prod$ is not complicated since we can calculate and store $P_{\underline{I}|H_1}$, $P_{\underline{I}|H_0}$, and $\prod$ once and then use them whenever needed.

- After calculating $P(D = 1|\underline{x}^q)$, P_e can be calculated. Note that the expression inside $\sum_{\underline{x}^q} \sum_{\underline{i}}$ has the same complexity as for calculating $P(D = 1|\underline{x}^q)$. Hence, the complexity in calculating P_e is approximately

$\underline{x}^q$ combinations number

$$\overbrace{\prod_{j=1}^{M} Q_j}^{M} \prod_{j=1}^{M} N_j \Big(\prod_{j=1}^{M} N_j \Big) = \prod_{j=1}^{M} Q_j \Big(\prod_{j=1}^{M} N_j \Big)^2. \tag{3.10}$$

As a special case, if all the sensors have the same number of subintervals and the same number of quantization levels (i.e $Q_1 = Q_2 = \cdots = Q_M = Q$ and $N_1 = N_2 = \cdots = N_M = N$), then the total fitness function complexity becomes $(QN^2)^M$. If we store in memory all the values of $P(D = 1|\underline{x}^q)$, then the fitness complexity reduces to $\Pi_{j=1}^{M} Q_j \Pi_{j=1}^{M} N_j \times (1) = \Pi_{j=1}^{M} Q_j \Pi_{j=1}^{M} N_j$, which equals to $(QN)^M$ in the given special case.

In the erroneous reports case: In a similar way as in the error-free case, we have the following.

- Calculating $P(D=1|\underline{x}^r)$ can be approximated by $\prod_{j=1}^{M} Q_j \prod_{j=1}^{M} N_j$ times.
- After calculating $P(D=1|\underline{x}^r)$, the P_e can be calculated. Hence, the complexity in calculating P_e is approximately

$$\prod_{j=1}^{M} Q_j \prod_{j=1}^{M} Q_j \prod_{j=1}^{M} N_j \Big(\prod_{j=1}^{M} Q_j \prod_{j=1}^{M} N_j \Big) = \Big(\prod_{j=1}^{M} Q_j \Big)^3 \Big(\prod_{j=1}^{M} N_j \Big)^2. \tag{3.11}$$

As a special case, if all the sensors have the same number of subintervals and the same number of quantization levels (i.e $Q_1 = Q_2 = \cdots = Q_M = Q$ and $N_1 = N_2 = \cdots = N_M = N$), then the total fitness function complexity becomes $(Q^3N^2)^M$. Also here, if we store in memory all the values of $P(D=1|\underline{x}^r)$, then the fitness complexity reduces to $\Pi_{j=1}^{M} Q_j \Pi_{j=1}^{M} Q_j \Pi_{j=1}^{M} N_j \times (1) = (\Pi_{j=1}^{M} Q_j)^2 \Pi_{j=1}^{M} N_j$, which equals to $(Q^2N)^M$ in the given special case.

3.3.2.2 GA Space Complexity

The GA memory requirement (i.e., space/storing complexity) is fairly low. This is due to the fact that the GA needs to save only one generation of solutions at each iteration (i.e., at any given time). If we assume that each solution and its corresponding fitness function value require one memory unit, then the space complexity of GA is $O(|\mathcal{G}|)$, where $|\mathcal{G}|$ is the total number of solutions in any given generation. Note that $|\mathcal{G}|$ is a constant value through the whole run of GA. If storing $P(D=1|\underline{x}^q)$, the space complexity will be increased while reducing the computational complexity.

3.4 Performance Evaluation

In this section, we provide simulations to study the performance of the GA procedure stated in the previous section and compare it with previously proposed procedures. The

simulations setup is presented first then the simulation results are presented under two separate cases, namely the error-free and erroneous reporting channels.

3.4.1 Simulations Setup

Because of its simplicity and popularity, the joint normal distribution is adopted in these simulations. P_e is taken as the performance metric. The performance of the proposed framework (termed as GAF[1]) is compared with the one proposed in [44] (termed as GSSA[1]) and the one proposed in [76] (termed as GBU[1]). Both the [44] and [76] frameworks consider only single-bit decisions from the sensors to the FC. In addition, the PBP algorithm is used in [44] and [76] to find a suboptimal solution. This algorithm suffers from the obtained solution being likely trapped at a local optimum. Another limitation of [44] and [76] is the use of specific FC rules, namely the majority logic (in GSSA), the AND, OR, and XOR (in GBU) fusion rules at the FC. None of these rules are optimal in all cases. That is, some changes in the system parameters can change the optimal fusion rule from one to another. On the contrary, GAF uses the LRT rule which is proven to be always optimal [63, 66].

The centralized detection performance (see Chapter 2 or[46]) is also plotted as the best performance possible (termed as Optimal). In the centralized detection, and since the normal distribution is considered in our simulations, the LRT and the P_e function can be derived as follows. The FC decides H_1 if

$$T \geq \gamma \equiv \frac{2\log\lambda + \mathcal{X}^T C_x^{-1}\mathcal{X}}{2} \tag{3.12}$$

and H_0 otherwise, where $T = Z^T C_x^{-1}\mathcal{X}$, $Z = [x_1\ x_2\ \ldots\ x_M]^T$, C_x is the covariance matrix under both hypotheses, $\mathcal{X} = V[\phi_1\ \phi_2\ \ldots\ \phi_M]^T$, and λ equals the ratio of prior probabilities, namely $P(H_0)$ divided by $P(H_1)$.

Let $W = C_x^{-1}\mathcal{X}$, then the probabilities of false alarm and miss detection are given as

1 GAF: Genetic Algorithm Framework, GSSA: Gauss-Seidel Sampling Algorithm framework, and GBU: Good Bad and Ugly framework.

$P_F = Q\left(\frac{\gamma}{\sqrt{W^T C_x W}}\right)$ and $P_M = Q\left(\frac{-\gamma + \mathcal{X}^T W}{\sqrt{W^T C_x W}}\right)$, respectively. $Q(\cdot)$ is the upper tail function of the standard Gaussian distribution.

Define $K = W^T C_x W$. Then the P_e with centralized detection is given as

$$\begin{aligned} P_e &= P(H_1)Q\left(\frac{-\gamma + \mathcal{X}^T W}{\sqrt{W^T C W}}\right) + P(H_0)Q\left(\frac{\gamma}{\sqrt{W^T C W}}\right) \\ &= P(H_1)Q\left(\frac{-2\log(\lambda) + K}{2\sqrt{K}}\right) + P(H_0)Q\left(\frac{2\log(\lambda) + K}{2\sqrt{K}}\right) \end{aligned}. \tag{3.13}$$

3.4.2 Simulations With Error-Free Reporting Channels

3.4.2.1 Comparison With the GBU Framework. Scenario 1:

We took the same system deployment in [76] for comparison. We use $M = 2$, $P(H_1) = \frac{2}{3}$, and the observations X_1 and X_2 at the sensors are normally distributed, under H_1 and H_0 as $(X_1, X_2) \sim \mathcal{N}(s_1, s_2, 1, 1, \rho)$ and $(X_1, X_2) \sim \mathcal{N}(0, 0, 1, 1, \rho)$, respectively. Where $\rho = 0.75$, $s_1 = \cos(\theta)$, $s_0 = \sin(\theta)$ and $\theta = [45^o, 135^o]$. For the GAF, we use the same system parameters besides having $Iter = 6000$ as the total number of iterations, the number of crossovers is $c = 2$, the number of subintervals per sensor is 10 (i.e., $N_1 = N_2 = 10$), total number of solutions per generation is $|\mathcal{G}| = 12$, the mutation rate is 5% (i.e., $\epsilon = 5$), and using three values for the quantization levels number (i.e., $Q \in \{2, 4, 8\}$). The results are given in Fig. 3.3(a), which shows the GAF with $Q = 2$ (i.e., GAF-$Q = 2$) outperforms both the AND and the XOR fusion rules and has the same performance as the OR fusion rule. We also notice that the GAF-$Q = 2$, the AND, the OR, and the XOR fusion rules have close performance for $45^o \leq \theta \leq 80^o$ and then the performance of the GAF-$Q = 2$ converges to the performance of XOR fusion rule which becomes the optimal fusion rule (using the GBU framework) in the interval $80^o \leq \theta \leq 135^o$. So that, using LRT in the GAF, the performance always converges to the optimal performance. Note that, in this scenario, the LRT may converge to a different fusion rules other than the XOR. This is because the GAF optimizes the local decisions and the fusion rule simultaneously, while the GBU optimizes the local decisions given the fusion

rule. Ultimately the GAF P_e performance converges to the optimal performance. To see the change in the GAF fusion rules with the change in θ, assume that $w_j = B(X_j^q - 1), \forall j = 1, 2$, where $B(x)$ is the binary representation of x, then the LRT in GAF-$Q = 2$ converges to the optimal fusion rules in Table 3.2 to provide the best P_e performance. Note that $\overline{w_j}$ is the compliment of w_j.

θ	45	50	55	60	65	70
FC rule	NAND	$w_1 + \overline{w_2}$	w_2	$\overline{w_2}$	w_2	w_2
θ	75	80	85	90	95	100
FC rule	$\overline{w_2}$	$\overline{w_2}$	$\overline{w_2}$	$\overline{w_1} + w_2$	$w_1 + \overline{w_2}$	NAND
θ	105	110	115	120	125	130
FC rule	OR	NAND	NAND	OR	$w_1 + \overline{w_2}$	OR

Table 3.2. GAF fusion rule Vs θ (in degrees). Here w_1 and w_2 are one-bit binary numbers.

The figure also shows that the use of $Q = \{4, 8\}$ gives better performance than that for the GBU framework and the GA with $Q = 2$. We also notice that as we increase Q, the resulting curves from the GA algorithm approach the optimal curve.

Scenario 2: In this scenario, we study the level of compliance of the GAF with the mathematical results presented in the GBU work (i.e., [76]). In the GBU work, the authors divided the (s_1, s_2) space into three regions: the "Good", the "Bad" and the "Ugly". In the "Good" region (i.e., when $(s_2 - \rho s_1)(s_1 - \rho s_2) \geq 0$ for positive ρ), it is always possible to quantize optimally at the sensors using a single-threshold to divide the local decision intervals at each sensor, [25]. The optimal fusion rule in this region is proved mathematically to be either AND or OR rule. In the "Bad" region (i.e., when the pair (s_1, s_2) is in the area between the line $s_2 = 0$ and $s_2 = \rho s_1$ or the area between the line $s_1 = 0$ and $s_2 = \frac{1}{\rho} s_1$ for positive ρ), there are two possibilities: first, a fusion decision rule that depends only on single sensor (with single-threshold) decision and ignore the other. Or second, at least one sensor has a nonsingle-threshold decision rule. The rest of the space is considered as an "Ugly" region (i.e., the second and the fourth quadrants of the (s_1, s_2) space when $\rho > 0$). For the the "Ugly" region, there is no clear rule provided in [76]. Next, we want to test the

compliance of the GAF-$Q = 2$ results with the mathematical results from [76] in the "Good" and "Bad" regions.

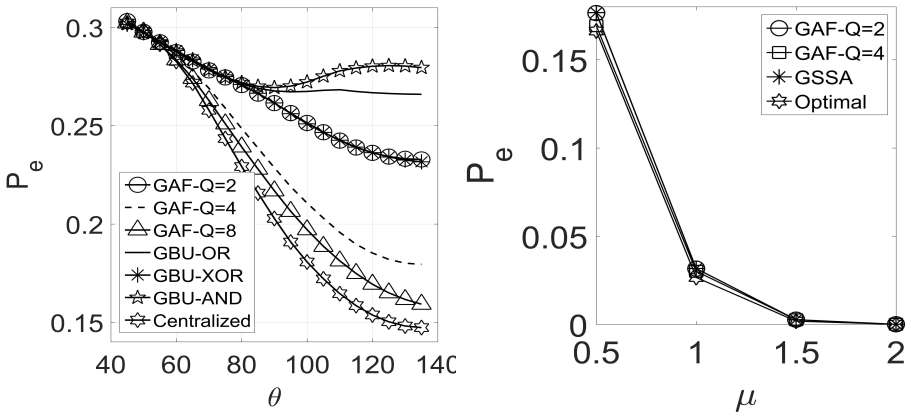
"Good" region example: In this example, $M = 2$, $P(H_1) = P(H_0) = 0.5$, $\rho = 0.75$, $Q_1 = Q_2 = 2$, $N_1 = N_2 = 7$, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $Iter = 2000$, $c = 2$ and to be in the "Good" region, we put $s_1 = s_2 = 2$ and $\sigma_1^2 = \sigma_2^2 = 1$. Now, by running the GA, we have the following results:

$$x_1^q = \Gamma_1(x_1) = \begin{cases} 1, \text{if } x_1 \in (0.705, \infty) \\ 2, \text{otherwise} \end{cases},$$

$$x_2^q = \Gamma_2(x_2) = \begin{cases} 1, \text{if } x_2 \in (-\infty, 0.705) \\ 2, \text{otherwise} \end{cases},$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q) \in \{(1,2)\} \\ 0, \text{otherwise} \end{cases}.$$

These results totally comply with the mathematical results in [76]. The local decisions are single-threshold decisions. And the FC fusion rule converged to an AND rule (using the LRT). To see the AND rule clearer, the fusion rule can be given as $D = \overline{w_1} w_2$ where $\overline{w_i}$ is the logical compliment of $w_i = B(X_i^q - 1)$, $w_i \in \{\text{"0"}, \text{"1"}\}$, and $B(X_i^q - 1)$ is the binary representation of $(X_i^q - 1), \forall i = 1, 2$. Note in this example, an alternative way to look at the fusion rule is that the set $\{(1,2)\}$ can be used to decide H_1 if the received pair (x_1^q, x_2^q) falls in it. Additionally, the FC can decide H_0, if the pair falls in the set $\{(1,1), (2,1), (2,2)\}$. Using these results from the GAF-$Q = 2$, the P_e at the FC is 0.14862. Using the exhaustive search (for the local thresholds at the sensors) with the LRT at the FC, the P_e value is 0.14859 with thresholds equal to 0.72 at both sensors. Indeed, the GA results are very close to the optimal values obtained by the exhaustive search.



(a) P_e Vs θ. (b) P_e Vs μ.

Figure 3.3. P_e Vs θ and the signal means.

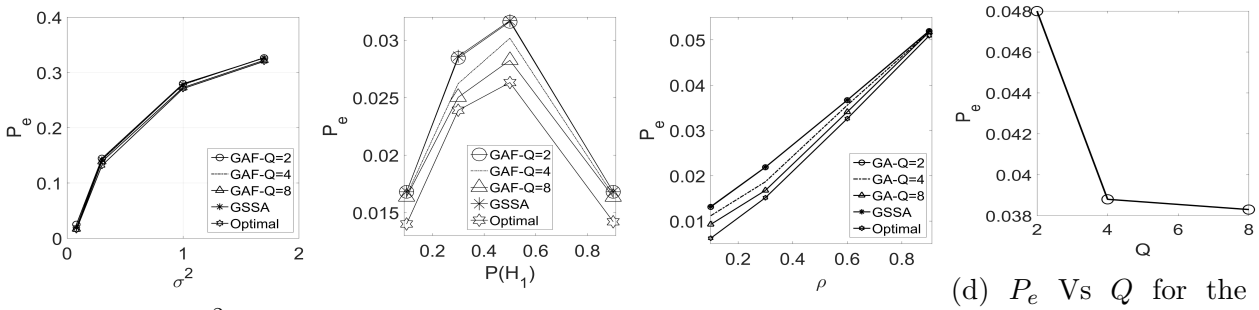
"Bad" region example: By changing s_2 to 1 in the previous example, the case falls in the "Bad" region. Hence, if we use the same system parameters but with $N_1 = N_2 = N = 6$, the results of the GAF-$Q = 2$ are as follows

$$x_1^q = \Gamma_1(x_1) = \begin{cases} 1, \text{if } x_1 \in (1, \infty) \\ 2, \text{otherwise} \end{cases},$$

$$x_2^q = \Gamma_2(x_2) = \begin{cases} 1, \text{if } x_2 \in (-\infty, 0.5) \cup (1, \infty) \\ 2, \text{otherwise} \end{cases},$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q) \in \{(1,1), (1,2)\} \\ 0, \text{if } (x_1^q, x_2^q) \in \{(2,1), (2,2)\} \end{cases}.$$

The results also match with mathematical results presented in [76], since the FC depends only on the decision coming from Sensor 1, which has a single-threshold decision regardless of Sensor 2's decision. Note that the fusion rule can be represented by $D = \overline{w_1}$. In this example, the P_e at the FC is 0.1587 and any change on $\Gamma_2(x_2)$ will not affect this P_e value at the FC.



(a) P_e Vs σ^2. (b) P_e Vs $P(H_1)$. (c) P_e Vs ρ. (d) P_e Vs Q for the cases of 2 sensors.

Figure 3.4. P_e Vs noise power, H_1 prior probability, correlation coefficient and quntization levels.

3.4.2.2 Comparison with GSSA

Comparisons with the GSSA are given in the following scenarios taken from [44].

Scenario 1: Assume $M = 3$, $Q_1 = Q_2 = Q_3 = Q \in \{2, 4\}$. Under H_1, the mean vector is $[\mu\ \mu\ \mu]$, and $C_x = \begin{bmatrix} \sigma^2 & \rho\sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \rho\sigma^2 & \sigma^2 \end{bmatrix}$, where $\rho = 0.5$ and $\sigma^2 = 0.1$. The covariance matrix under H_0 is the same as the one under H_1 while the mean vector here is [0 0 0]. We assume $P(H_1) = P(H_0) = 0.5$, and $N_1 = N_2 = N_3 = N = 4$. This means that $\zeta_j = 0.25, \forall j = \{1, 2, 3\}$. We also assume $|\mathcal{G}| = 6$, $Iter = 3200$, $c = 3$, and $\epsilon\% = 5\%$. Fig. 3.3(b) plots the P_e with respect to μ that changes from 0.5 to 2. As we can see from Fig. 3.3(b), the GAF with $Q = 2$ achieves the same performance as the GSSA. But the GAF with higher Q value has better performance. We notice that the GAF with $Q = 4$ does not give a significant improvement over that with $Q = 2$. This is because the GSSA and the GAF with $Q = 2$ in this scenario provide performance already very close to the optimal performance, which leaves a tiny room for the GAF with higher Q values to further improve the performance. We also notice that all the P_e curves decrease with respect to μ as increasing μ increases the signal-to-noise-ratio (SNR) at the sensors, hence, gives more accurate local decisions are sent to the FC.

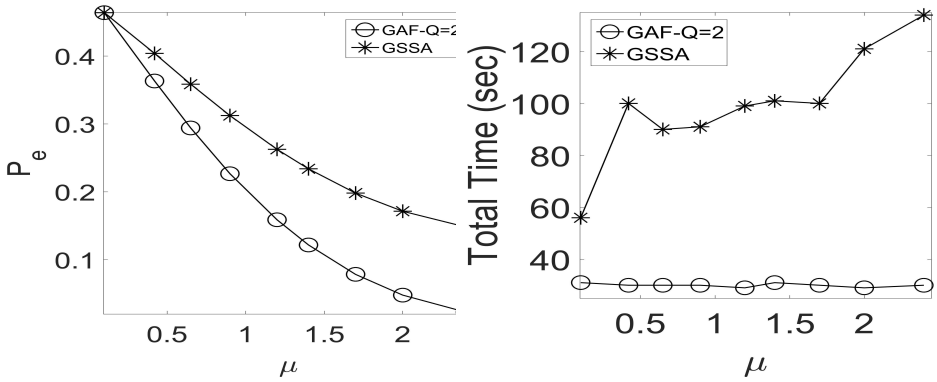
Scenario 2: In this scenario, we have the same system characteristics as in Scenario 1 mentioned above but with $N = 10$ (i.e., $\zeta_j = 0.1, \forall j = \{1, 2, 3\}$), fixing μ to 1, and changing σ^2 from 0.08 to 1.7. Fig. 3.4(a) plots the P_e as a function of σ^2 for the GSSA, the GAF

with $Q \in \{2, 4, 8\}$, and the centralized detection case. For all the methods, we can see the increase in the P_e value as σ^2 increases. As in Fig. 3.3(b), the performances of GSSA and GAF with $Q = 2$ are the same and very close to the optimal values. As a result, the GAF with $Q \in \{4, 8\}$ does not have much room to provide better performance.

Scenario 3: Here, the same system parameters are as those in Scenario 1 but with $N = 10$, $\mu = 1$, and $P(H_1)$ changes from 0.1 to 0.9. This scenario studies the P_e Vs $P(H_1)$. Fig. 3.4(b) plots the P_e Vs $P(H_1)$ for GSSA, GAF, and the optimal frameworks. The figure shows that for all methods the highest P_e value happens at $P(H_1) = 0.5$. This is because at $P(H_1) = 0.5$ the most uncertainty happens about which hypothesis generated the vector $Z = [x_1, x_2, \ldots, x_M]$. As we can see the P_e performances of both the GSSA and the GAF with $Q = 2$ are the same, we also notice that the GAF with higher Q values improves the performance.

Scenario 4: This scenario studies the P_e performance against the increase in the correlation coefficient (i.e., ρ). The same system parameters of Scenario 1 are used here but with $N = 10$, $\mu = 1$, $Iter = 8000$, and ρ changes from 0.1 to 0.9. As it was proven in [46], for the case of the centralized detection and when all received signals' means are equal, the P_e increases with the increase in ρ. The behavior of the decentralized curves is similar to the case of the centralized detection [i.e., "Optimal" curve in Fig. 3.4(c)]. It is also noticeable that by increasing the number of quantization levels, the P_e curves get closer to the optimal one. Note that the case of having unequal received signals mean values has different results and is considered later.

Fig. 3.4(d) plots the P_e performance of the GAF as a function of quantization levels number per sensor (Q). This figure shows the P_e value decreases as Q increases. This is predictable, since each sensor sends higher amount of information as Q increases. Hence, more accurate global decision can be made at the FC. The results in this figure are generated by considering the system with the parameters: $M = 2$, $Q_1 = Q_2 = Q \in \{2, 4, 8\}$, $N_1 = N_2 = N = 10$, $|\mathcal{G}| = 24$, and $Iter = 6000$.



(a) P_e Vs μ. (b) Time Vs μ.

Figure 3.5. P_e and Time complexity Vs μ in Scenario 5.

Scenario 5: As mentioned previously, the majority logic rule used in GSSA is not optimal in all scenarios. Hence, this scenario gives one of those cases where the GSSA is outperformed (in terms of P_e) by GAF using binary decision rules at the sensors. Assume that $M = 3$, the signal mean vector under H_1 is $\underline{\mu} = [0.1\ \mu\ 0.1]$ with μ changes from 0.1 to 2.4, and is zero vector under H_0, use equicorrelated covariance matrix under both H_1 and H_0 with $\sigma_1 = \sigma_2 = \sigma_3 = 0.6$ and $\rho = 0.7$, $P(H_1) = P(H_0) = 0.5$, $N_1 = N_2 = N_3 = 4$, $\epsilon\% = 5\%, |\mathcal{G}| = 6$, $c = 3$, and $Iter = 3200$. Fig. 3.5(a) plots the P_e of both GAF-$Q = 2$ and GSSA with respect to μ, as it is obvious the GSSA is outperformed by the GAF, this is because as μ increases the decision effect of Sensor 1 and Sensor 3 become insignificant compared to the decision coming from Sensor 2. Hence, the LRT in GAF treats the decision coming from Sensor 2 with more consideration compared to the decisions coming from Sensor 1 and Sensor 2. When μ is larger on the other hand, the majority logic in GSSA treats all the decisions coming from all the sensors with the same importance. As a result, the GAF provides better performance. Fig. 3.5(b) plots the time consumption of both the GAF-$Q = 2$ and GSSA in this scenario. This figure shows that the GSSA is also outperformed by the GAF-$Q = 2$ in terms of the time complexity. It should be noted that the time complexity in GAF does not depend on the μ value [which is clear from Equation (3.10) and the figure], while the GSSA time complexity increases with μ (see [44]). More comments on the time complexity comparison between the GAF and GSSA are provided later.

Fig. 3.7(a) plots the GAF P_e performance as a function of M. These results are generated using the system parameters of $Iter = 2000$, $M \in \{1, 2, 3, 4, 5, 6, 7\}$, the same quantization levels at all the sensors with $Q = 2$, $\rho = 0.5$, $\sigma^2 = 0.1$. The signal mean value at each sensor under H_1 is 1 and 0 under H_0, $P(H_1) = 0.5$, $N_1 = N_2 = N = 4$, $|\mathcal{G}| = 12$ and $\epsilon\% = 5\%$. The figure shows that the increase in the number of sensors provides higher amount of information at the FC, hence decreasing the P_e value.

3.4.2.3 Different Received Signal Means:

In the previous GSSA Scenario 4 simulations, we studied the effect of ρ on P_e at the FC when the received means are all equal. In this part of simulations, we study the effect of ρ on the P_e at the FC when the received means are different. Assume $M = 3$, and the mean values under H_1 are $\underline{\mu}_1 = [1, 5, 3]$ and under H_0 are $\underline{\mu}_0 = [0, 0, 0]$. The equicorrelated covariance form is used with $\sigma^2 = 0.5$ and $\frac{-1}{2} < \rho < 1$. We also assume $P(H_1) = 0.5$, $N = 8$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $Iter = 8000$ and $c = 3$. In Fig. 3.7(d), The P_e curves are plotted for different values of Q, where $Q_1 = Q_2 = Q_3 = Q$ and $Q \in \{2, 4, 8\}$. The P_e of the Optimal centralized detection case is also considered.

For our example, when $\rho = \rho^* = 0.3476$, P_e reaches the maximum for the optimal curve [46]. The figure also shows that the other DD curves (when $Q = 2, 4, 8$) follow similar behavior as the centralized counterpart. Interestingly, the local decisions when $\rho \to 1$ follow closely to those presented in Theorem 1. It is worth mentioning that we tested the P_e behavior at the FC with respect to ρ while having AND fusion rule in a two-sensor system with different signal means. The results for this case does not follow the same behavior as the LRT shown in this section.

3.4.3 GA physical limitations:

Increasing N increases the computations time. To elaborate, with $N = 4$ and $M = 7$, we have $N^M = 16384$ sub-spaces that needs a couple of days to get the final solution. On the other hand, it is hard to find the exact complexity of the PBP algorithm (i.e., the number

of times of calculating the P_e function and the time complexity in calculating P_e) in [44] because it depends on multiple multivariate nonlinear functions that keep changing in every iteration. This change is untraceable, which makes it hard to predict the exact iteration at which the algorithm converges to the final answer. But from the presented simulations for GSSA in [44], it can be seen that at most 5 sensors were used, which gives a hint about the huge complexity in the PBP algorithm. Fig. 3.5(b) shows that the GSSA has higher complexity than the GA while the later has better P_e performance. In another example, if we consider Scenario 1 [in Fig. 3.3(b)], we have the time complexities of the GSSA and the GA given in Table 3.3. Note that the GA with $N = 4$ gives the same performance as the GSSA but with lower complexity. Now, if we increase N to 10 in the GA, the complexity of GA becomes higher than the GSSA. In terms of the P_e, the GA with $N = 10$ performance is within $\pm 10^{-4}$ range off the performance of GSSA and GA with $N = 4$. This difference is negligible and for the sake of lower complexity we can choose $N = 4$ in this example.

μ	0.5	1	1.5	2
GSSA	52	97	92	126
GAF-$Q = 2$ and $N = 4$	30	32	31	30
GAF-$Q = 2$ and $N = 10$	869	1016	885	1114

Table 3.3. Time complexity (in sec) of GAF-$Q = 2$ Vs GSSA in Scenario 1.

To clarify the complexity in terms of the total time spent by the GA. Fig. 3.7(b) plots the total time spent by the GA with the number of subintervals (N) per sensor. This figure is generated using a sensor system with the parameters: $M = 2$, $Iter = 3200$ for the N ($N = N_1 = N_2$) values from 2 to 19 and $Iter = 6000$ for N values from 20 to 23, the quantization levels at all the sensors are the same $Q = Q_1 = Q_2 = 2$, the received signals at the sensors are normally distributed with $\begin{bmatrix} \frac{5}{6} & 0.675 \\ 0.675 & \frac{4}{3} \end{bmatrix}$ covariance matrix under both H_1 and H_0, the signal mean vector at the sensors is $[1, \frac{3}{4}]$ under H_1 and $[0, 0]$ under H_0, $P(H_1) = \frac{2}{3}$, $|\mathcal{G}| = 12$, and $\epsilon\% = 5\%$. As it is clear from Fig. 3.7(b), the total elapsed time by

the GA increases very sharply as the N increases by small values. Equation (3.10) calculates the time complexity value needed for every fitness function calculation.

How to deal with this complexity limitation?

For this system parameters, Fig. 3.7(c) plots the GAF P_e curve with N. As it clear from the figure, for very large N values, the GA accuracy increases. This is because the larger the N the higher the number of subintervals on the sensing axis. The edges of the subintervals are considered as possible thresholds in the final sensor decision rule. Hence, there is a higher chance some of these thresholds come close to the optimal values. On the other hand, if N is small, it still has a chance to provide a smaller P_e performance to a larger N. This is because $N1$ could provide few thresholds but closer to the optimal thresholds, while the thresholds provided by the larger N might be farther away from the optimal ones; This is clear from the figure, where $N = 3$ provides P_e performance as good as the $N = 20$ P_e and better than $N = 7$. Therefore, even though large values of N provide accurate GA results, the higher the N the slower the GA. Hence, there is a tradeoff between the speed and accuracy, the choice depends on the problem specifications. For the example in Fig. 3.7(c), taking $N = 3$ provides the same P_e performance as using higher N values. On the other hand, using the value 3 saves a huge amount of time compared to the case of using 20 (for example) as it is clear in Fig. 3.7(b). It is worth mentioning that increasing the number of subintervals has another downside in addition to the complexity. This is due to the fact that increasing the number of subintervals means means increasing the number of variables to be optimized in the solution; hence, the algorithm may need higher number of iterations to reach the optimality compared to the small number of subintervals that needs less number of iterations and gives the same P_e value. As a result, for the same example in Fig. 3.7(c), the number of iterations is 3200 for the N values from 2 to 19 and 6000 for the N values from 20 to 23. This means just by using 3200 iterations with $N = 3$, we can get the same performance of having 6000 iterations and $N = 20$ with much less time as it is shown in (3.10) and Fig. 3.7(b).

3.4.3.1 GAF adaptability

In this part of the simulations, we want to show how the GAF can update the local decisions at the sensors and the fusion rule simultaneously, in order to minimize the P_e at FC if a change happens to one of the system parameters.

EXAMPLE 1 The studied system here has the following parameters: $M = 3$, $Q_1 = Q_2 = Q_3 = 2$, $\underline{\mu}_1 = [1, 2, 1.5]$, $\underline{\mu}_0 = [0, 0, 0]$, the equicorrelated covariance matrix is adopted with correlation coefficient ρ and $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = 0.5$, $P(H_1) = P(H_0) = 0.5$, $N = 6$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $c = 3$, and $Iter = 8000$. At first, we want to find the results of GAF when $\rho = 0.3$, then find the results of GAF if we just change ρ to 0.8.

When $\rho = 0.3$ the results are:

$$x_1^q = \Gamma_1(x_1) = \begin{cases} 1, \text{if } x_1 \in (0.5, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$x_2^q = \Gamma_2(x_2) = \begin{cases} 1, \text{if } x_2 \in (1, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$x_3^q = \Gamma_3(x_3) = \begin{cases} 1, \text{if } x_3 \in (1.2655, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q, x_3^q) \in \{(1,1,1), (1,1,2), (1,2,1), (2,1,1), (2,1,2)\} \\ 0, \text{if } (x_1^q, x_2^q, x_3^q) \in \{(1,2,2), (2,2,1), (2,2,2)\} \end{cases}.$$

$$P_e = 0.0746.$$

When $\rho = 0.8$ the results are:

$$x_1^q = \Gamma_1(x_1) = \begin{cases} 1, \text{if } x_1 \in (-0.3587, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$x_2^q = \Gamma_2(x_2) = \begin{cases} 1, \text{if } x_2 \in (1, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$x_3^q = \Gamma_3(x_3) = \begin{cases} 1, \text{if } x_3 \in (0.75, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q, x_3^q) \in \left\{ \begin{matrix} \{(1,1,1),(1,1,2),(1,1,2),(2,1,1),(2,1,2), \\ (2,2,1) \end{matrix} \right\} \\ 0, \text{if } (x_1^q, x_2^q, x_3^q) \in \{(1,2,1),(1,2,2),(2,2,2)\} \end{cases}.$$

$P_e = 0.07827.$

Fig. 3.6 visualizes the local decisions at each sensor (in Example 1 when $\rho = 0.3$) along with the corresponding pdfs. The local decisions when $\rho = 0.8$ can be visualized in a similar manner.

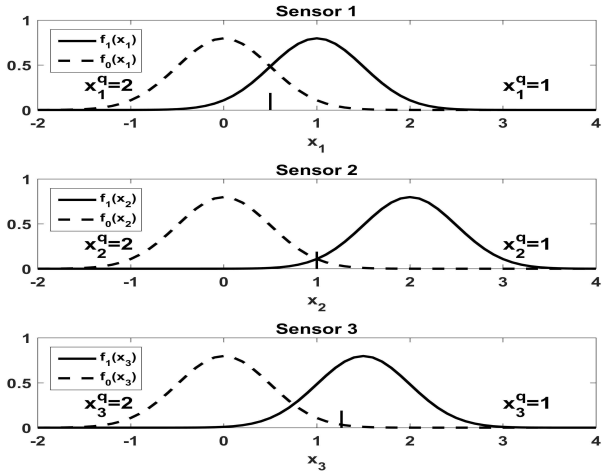
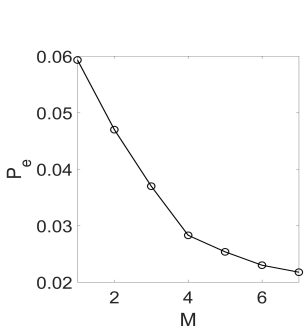
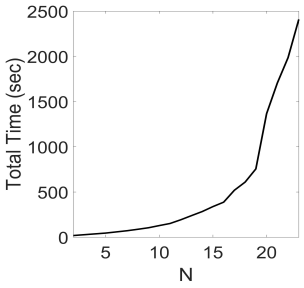


Figure 3.6. Example 1 local decisions visualization when $\rho = 0.3$

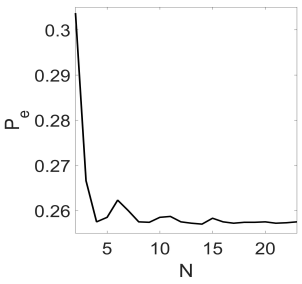
Example 2: Assume a system consists of $M = 3$ sensors, $\mu_1 = [1, 5, 3]$, $\mu_0 = [0, 0, 0]$, equicorrelated covariance matrix with $\sigma^2 = 0.5$ and $\rho = 0.5$, $P(H_1) = 0.5$, $N = 8$, $|\mathcal{G}| = 6$,



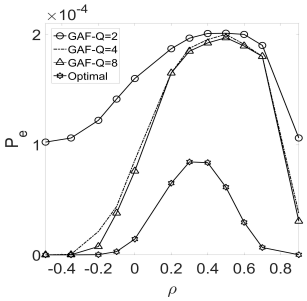
(a) P_e Vs M.



(b) Time Vs N for the cases of 2 sensors.



(c) P_e Vs N for the cases of 2 sensors.



(d) P_e Vs ρ for the cases of 3 sensors and different means.

Figure 3.7. P_e Vs number of sensors, Time and P_e Vs number of sub-intervals and P_e Vs correlation coefficient.

$Iter = 8000$, and $c = 3$.

In the case when $Q = 2$ quantization levels per sensor, we have:

$$x_1^q = \begin{cases} 1, if x_1 \in (-\infty, 1.1051) \\ 2, \text{otherwise} \end{cases}.$$

$$x_2^q = \begin{cases} 1, if x_2 \in (2.5, \infty) \\ 2, \text{otherwise} \end{cases}.$$

$$x_3^q = \begin{cases} 1, if x_3 \in (-\infty, 0.4765) \\ 2, \text{otherwise} \end{cases}.$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q, x_3^q) \in \{(1,1,1),(1,1,2),(2,1,2)\} \\ 0, \text{if } (x_1^q, x_2^q, x_3^q) \in \{(1,2,1),(1,2,2),(2,1,1),(2,2,1),(2,2,2)\} \end{cases}.$$

$P_e = 2.010453 \times 10^{-4}$.

In the case when $Q = 4$ quantization levels per sensor, we have:

$$x_1^q = \begin{cases} 1, & \text{if } x_1 \in (-\infty, -0.1051) \\ 2, & \text{if } x_1 \in (-0.1051, 0.7885) \\ 3, & \text{if } x_1 \in (1.1051, 1.5135) \\ 4, & \text{otherwise} \end{cases}.$$

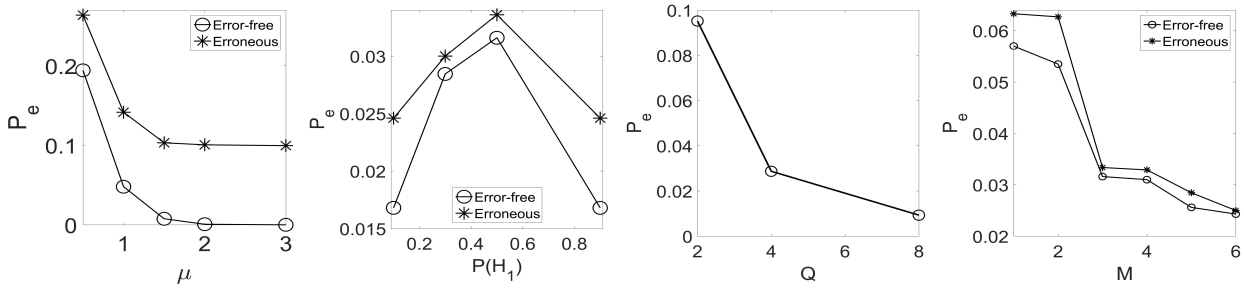
$$x_2^q = \begin{cases} 1, & \text{if } x_2 \in (4.5231, 5) \\ 2, & \text{if } x_2 \in (-\infty, 2.5) \\ 3, & \text{if } x_2 \in (2.5, 4.5231) \\ 4, & \text{otherwise} \end{cases}.$$

$$x_3^q = \begin{cases} 1, & \text{if } x_3 \in (-\infty, -0.4769) \cup (0, 0.4765) \\ 2, & \text{if } x_3 \in (2.5235, 3.4769) \\ 3, & \text{if } x_3 \in (0.4765, 1.5) \\ 4, & \text{otherwise} \end{cases}.$$

$$D = \begin{cases} 1, \text{if } (x_1^q, x_2^q, x_3^q) \in \begin{cases} (1,1,1),(1,1,2),(1,1,3),(1,1,4), \\ \qquad (1,2,2), \\ (1,3,1),(1,3,2),(1,3,3),(1,3,4), \\ (1,4,1),(1,4,2),(1,4,3),(1,4,4), \\ (2,1,1),(2,1,2),(2,1,3),(2,1,4), \\ (2,3,1),(2,3,2),(2,3,3),(2,3,4), \\ (2,4,1),(2,4,2),(2,4,3),(2,4,4), \\ (3,1,1),(3,1,2),(3,1,3),(3,1,4), \\ \qquad (3,3,2),(3,3,3),(3,3,4), \\ (3,4,1),(3,4,2),(3,4,3),(3,4,4), \\ (4,1,1),(4,1,2),(4,1,3),(4,1,4), \\ (4,3,1),(4,3,2),(4,3,3),(4,3,4), \\ (4,4,1),(4,4,2),(4,4,3),(4,4,4) \end{cases} \\ 0, \text{if } (x_1^q, x_2^q, x_3^q) \in \begin{cases} (1,2,1), \qquad\qquad (1,2,3),(1,2,4), \\ (2,2,1),(2,2,2),(2,2,3),(2,2,4), \\ (3,2,1),(3,2,2),(3,2,3),(3,2,4), \\ (3,3,1), \\ (4,2,1),(4,2,2),(4,2,3),(4,2,4)\} \end{cases} \end{cases}.$$

$P_e = 1.988153 \times 10^{-4}$.

Fig. 3.8 visualizes the local decisions at each sensor (in Example 2 when $Q = 4$) along with the corresponding pdfs. The local decisions when $Q = 2$ can be visualization in a similar manner.



(a) P_e Vs μ. (b) P_e Vs $P(H_1)$. (c) P_e Vs Q. (d) P_e Vs M.

Figure 3.9. P_e Vs signal means, H_1 prior probability, quantization levels and number of sensors.

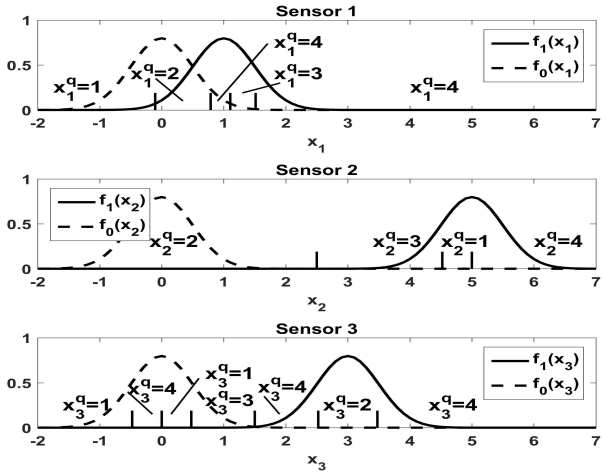


Figure 3.8. Example 2 local decisions visualization when $Q = 4$

As we can see from these examples, the GAF can adapt the local decisions at the sensors along with the fusion rule at the FC, simultaneously, in order to minimize the P_e at the FC if a change occurs to the system parameters. This is also clear in the examples (given previously) on testing the agreement of the GAF with the mathematical results in [76] when the change happened on the signal mean s_1 from 2 to 1. Clearly, if the quantization level and the number of sensors increase, the possible number of fusion rule increases quickly causing prohibited high complexity.

3.4.3.2 Prior probability mismatch

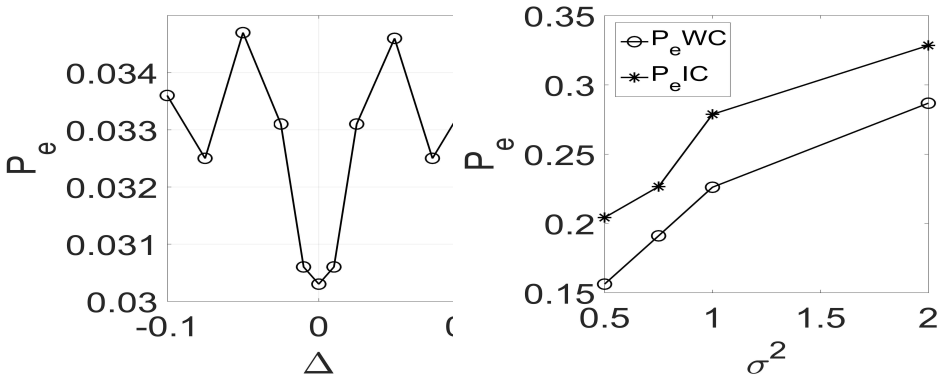
Even though the proposed work assumes perfect knowledge of the prior probabilities (i.e., $P(H_0)$ and $P(H_1)$), it is interesting to test the robustness of the proposed framework

against the mismatch between the assumed/estimated prior probabilities (used in GA to generate the local and fusion decisions) and the real unknown prior probabilities. To this end, assume the following system setup: $M = 3$, $Q_1 = Q_2 = Q_3 = 4$, $\underline{\mu}_1 = [1, 1, 1]$, $\underline{\mu}_0 = [0, 0, 0]$, the equicorrelated covariance matrix is adopted with $\rho = 0.5$ and $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = 0.1$, $N = 4$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $c = 3$, $Iter = 3200$, and the real H_0 prior probability is $P(H_0) = 0.5$. Using this system setup, Fig. 3.10(a) plots the P_e value against Δ, where the mismatch value is $\Delta = P^*(H_0) - P(H_0)$, $P^*(H_0)$ is the estimated H_0 prior probability used by GA to generate the local and fusion rules, and $P(H_0)$ is the actual value of H_0 prior probability. $P^*(H_1)$ is just $1 - P^*(H_0)$. Note from the figure that the P_e has its lowest value at $\Delta = 0$, which is expected, since the estimation of $P(H_0)$ is perfect when $\Delta = 0$. From the figure, when $|\Delta| = 0.1$ (i.e., the $P^*(H_0) = 0.4$ or 0.6), the estimation error in $P(H_0)$ (which equals 0.5 in this example) is 20%. The corresponding P_e increased at this estimation when compared to the P_e at $\Delta = 0$ (i.e., perfect estimation). This percentage increase is 10.9. This example shows a good robustness feature of the proposed framework.

The symmetry in the figure comes from the fact that readings at the sensors have the same correlation structure (i.e., covariance matrix) under both hypotheses. The other behavior that can be noticed in the figure is the decrease in P_e value as $|\Delta|$ increases from 0.05 to 0.075. Even though it is hard to explain this behavior exactly due to the intractable nature of the suboptimal solution. It could also be attributed to the nonlinear nature of the system, which means some solutions may be perfect for different values of prior probabilities. For example the GA solution with $P(H_0) = 0.6$ and the GA solution with $P(H_0) = 0.3$ may be the same or very similar. This means, designing a GA solution with $P^*(H_0) = 0.6$ while the real $P(H_0) = 0.3$ (i.e., $\Delta = 0.3$ with 100% estimation error), may not affect the value of the real P_e.

3.4.3.3 Effect of ignoring the correlation

This section presents a scenario showing how important to consider the correlation in designing local and fusion decision rules. The system setup is as follows: $M = 3$, $Q_1 =$



(a) P_e Vs Δ (b) P_e Vs σ^2.

Figure 3.10. P_e Vs Δ, P_e Vs σ^2 with and without correlation consideration.

$Q_2 = Q_3 = 4$, $\underline{\mu}_1 = [1, 0.75, 0.5]$, $\underline{\mu}_0 = [0, 0, 0]$, the covariance matrix has the structure $C_x = \begin{bmatrix} \sigma^2 & 0.75\sigma^2 & 0.95\sigma^2 \\ 0.75\sigma^2 & \sigma^2 & 0.55\sigma^2 \\ 0.95\sigma^2 & 0.55\sigma^2 & \sigma^2 \end{bmatrix}$, σ^2 changes from 0.5 to 2, $N = 8$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $c = 3$, $Iter = 6000$, and $P(H_0) = 0.5$. Using this system setup, Fig. 3.10(b) plots the P_e Vs σ^2 when correlation is taken into consideration (i.e., using C_x in designing the decision rules at the sensors and the FC) and under the case of ignoring the correlation (i.e., using the covariance matrix $\sigma^2\mathbf{I}$ to design the decision rules, where $\mathbf{I}$ is an $M \times M$ identity matrix). The P_e with the correlation taken into consideration is termed by P_eWC, while the P_e with no consideration for correlation is termed P_eIC. As the figure shows, ignoring the correlation can reduce the distributed detection performance significantly (up to 31% in this example).

3.4.4 Simulation with Erroneous Reporting Channels

This part studies the effect of the reporting channels' errors on the system P_e performance at the FC.

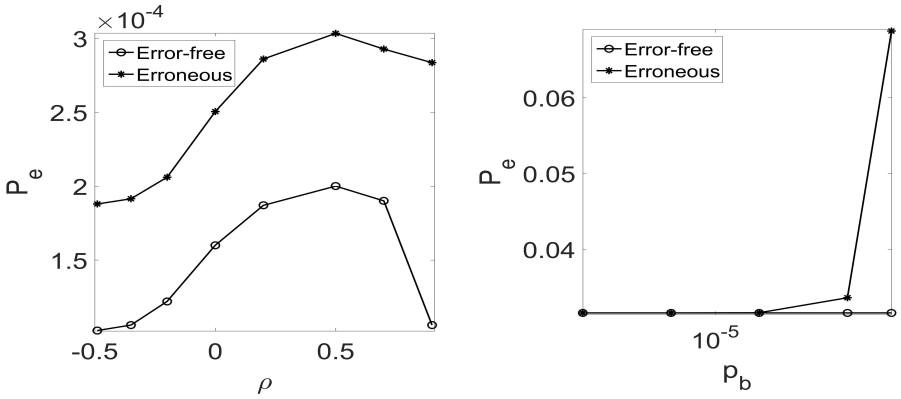
Scenario 1: This scenario studies the P_e at the FC against the mean received at each sensor under H_1 (i.e., μ), assuming that $\mu_1 = \cdots = \mu_M = \mu$. Let $M = 2$, $Q = 2$ per sensor, the received means under H_0 are all zeros, the covariance matrix (under both H_1 and H_0) has the equicorrelated model with $\sigma^2 = 0.1$ and $\rho = 0.5$, $N = 10$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 20$, $Iter = 300$, $c = 4$, $P(H_1) = 0.5$, and $P_{b1} = P_{b2} = 0.1$. By using the GA, Fig.

3.9(a) plots the P_e value at the FC for both the error-free and erroneous reporting channels cases. As expected, the P_e decreases with the increase in μ because of the increase in the SNR. Due to errors on the reporting channels, the system performs poorly (in terms of the P_e) compared to the error-free reporting channels case. We also notice that by increasing the μ the P_e curve for the erroneous channels case reaches the 0.1 limit (i.e., the bit error on each reporting channel). Any further increase in μ is not going to enhance the performance. On contrary, the P_e curve of the error-free channels keeps decreasing as μ increases.

Scenario 2: In this scenario, we study the P_e as a function of $P(H_1)$. The parameters used here are: $M = 3$, $Q_1 = Q_2 = Q_3 = 2$, under H_1: $\mu_1 = \mu_2 = \mu_3 = 1$, under H_0: $\mu_1 = \mu_2 = \mu_3 = 0$, the equicorrelated covariance matrix is adopted with $\sigma^2 = 0.1$ and $\rho = 0.5$, $N_1 = N_2 = N_3 = 10$, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $Iter = 8000$, $c = 3$, and $P_{b1} = P_{b2} = P_{b3} = 0.01$. Fig. 3.9(b) plots the P_e curves against $P(H_1)$ and for both cases (i.e., the error-free erroneous reporting channels), the P_e has the highest value at $P(H_1) = 0.5$. As expected, the curve of the erroneous reporting channels is worse than the error-free channels curve.

Scenario 3: In this scenario, we study the P_e as a function of the number of quantization levels Q per sensor. The parameters used here are: $M = 2$, under H_1: $\mu_1 = \mu_2 = 2$, under H_0: $\mu_1 = \mu_2 = 0$, the equicorrelated covariance matrix is adopted with $\sigma^2 = 0.1$ and $\rho = 0.5$, $N_1 = N_2 = N_3 = 10$, $\epsilon\% = 5\%$, $|\mathcal{G}| = 4$, $Iter = 6000$, $c = 2$, and $P_{b1} = P_{b2} = P_{b3} = 0.1$. Fig. 3.9(c) plots the P_e at the FC as a function of Q. As expected, increasing the quantization levels (i.e., number of reporting bits) reduces the P_e because of the increase in the information amount being transmitted to the FC.

Scenario 4: This scenario studies the P_e as a function of the number of the sensors (M). The parameters used here are: $Q = 2$ per sensor, under H_1: the received means are 1s, under H_0: the received means are 0s, the equicorrelated covariance matrix model is used with $\sigma^2 = 0.1$ and $\rho = 0.5$, $N = 4$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 12$, $Iter = 2000$, $c = 2 \times M$, and the probability of bit error is 0.01 per reporting channels. Fig. 3.9(d) shows the error-free curve performs better than the erroneous curve, but both curves decrease with the increase



(a) P_e Vs ρ. (b) P_e Vs bit error probability.

Figure 3.11. P_e Vs correlation coefficient and probability of bit error.

in the number of sensors.

Scenario 5: In this scenario, we study the P_e at the FC as a function of the correlation coefficient (ρ) while considering the reporting error. The parameters used here are as follows: the number of sensors is $M = 3$, $Q = 2$ per sensor; under H_1, the received means are $[1, 5, 3]$; under H_0, the received means are 0s; the equicorrelated covariance matrix model is used with $\sigma^2 = 0.5$ and $-0.5 < \rho < 1$, $N = 8$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $Iter = 8000$, $c = 3$, $P(H_1) = 0.5$, and the probability of bit error is 10×10^{-4} per reporting channels. Fig. 3.11(a) plots the P_e curves of the erroneous reporting case and the error-free reporting case against ρ. As seen, the erroneous case has worse performance compared to the error-free reporting case. We also notice that both curves follow the behavior of the centralized detection (i.e., Optimal case) explained earlier for Fig. 3.7(d).

Scenario 6: Finally, we study the P_e at the FC as a function of the probability of bit error on the reporting channels. The parameters used here are: $M = 3$, $Q = 2$ per sensor, under H_1: the received means are $[1, 1, 1]$, under H_0: the received means are 0s, the equicorrelated covariance matrix model is used with $\sigma^2 = 0.1$ and $\rho = 0.5$, $N = 8$ per sensor, $\epsilon\% = 5\%$, $|\mathcal{G}| = 6$, $Iter = 4000$, $c = 3$, $P(H_1) = 0.5$, and the probability of bit error is p_b per reporting channel. The p_b changes between 10^{-8} to 10^{-1}. Fig. 3.11(b) is a semi-logarithmic plot showing the P_e increasing as p_b increasing, which is expected. Also, as the p_b decreases, the P_e curve approaches the value of the error-free case.

3.5 Conclusion

This chapter has proposed a soft-decision-based DD system framework that considers correlation among the sensors' readings. The detection problem has been formulated as an INLP problem. As a result, it can find the local soft-decision rules at the sensors and the optimal fusion rule at the FC simultaneously to minimize the P_e at the FC. Both cases of error-free and erroneous reporting channels are considered. The GA-based solution proposed here can handle moderate number of subspaces and quantization levels with moderate number of sensors at reasonable computational complexity.

CHAPTER 4

Decentralized Detection of Signals with Joint Clayton Copula Distribution Model

This chapter considers a decentralized detection system for a binary hypothesis testing problem. The sensor signals are marginally Gaussian distributed with θ-parameter Clayton copula joint distribution. Boolean functions of two variables and likelihood ratio test at the fusion center are considered to generate binary global decisions. The exhaustive search and the genetic algorithms are proposed to provide the local sensor decision rules in each scenario such that the probability of error at the fusion center is minimized. The main purpose of this chapter is to study the probability of error behavior of the proposed detection systems in a correlated environment characterized by a joint Clayton copula distribution. The results show that in a correlated environment a sound detection system should be characterized by the likelihood ratio test at the fusion center and multiple-threshold decisions at the sensors.

4.1 Introduction

As has been stated previously, detection is an important tool in many real life applications, to name a few: target detection in radar systems, signal detection in cognitive radio networks and wireless networks in general. It can be a primary step towards estimation, or it can be the sole purpose of the sensor system as in surveillance [21, 46]. Detection is also needed in space exploration and deciding the existence of objects/phenomena. It plays a vital role in the problem of LTE an WiFi coexistence on the unlicensed frequency spectrum. Detection enables each network to detect the transmissions of the other. This information is then handed up to the MAC layer in order to take the necessary decision of deciding which channel to access based on the network regulations and cell transmission requirements.

The main downside of all the previously mentioned works, in previous chapters, is the assumption of the full knowledge about the joint probability density function (pdf) of the observations at the sensors. Since having full knowledge about the joint pdf is a complex problem, the work in [42] addressed this issue by providing a framework to estimate the joint pdf of complexly correlated observations using the copula theory. The authors provided several joint pdf models and addressed the problem of identifying the best copula model that fits the provided correlated data.

In this chapter, we are interested in distributed detection systems with dependent observations at the sensors. The goal is to find the local decision rules at the sensors to achieve the minimum P_e at the FC. A study of four fusion rules at the FC (namely: AND, OR, XOR, and LRT) is provided. In this work we apply some results provided by the copula theory in modeling unknown joint pdfs. More specifically, we are interested in studying the system performance of the four previously mentioned fusion rules while assuming the Clayton copula model. For Clayton copula model, we are using the Exhaustive Search and GA-based [45, 48] methods to find the local sensor decisions to minimize the P_e at the FC. This work can also be helpful in spaceflight extreme environments, where correlated measurements are taken with unknown pdfs.

4.2 Network model and Problem Formulation

In this chapter, we are interested in the system model given in Figure 4.1. This system consists of a primary radio user (PU), two sensors, and a FC. The sensors have no connection among them and their sole purpose is to sense the received signal from the PU, quantize it using their local decision rules (i.e. $\Gamma_1(\cdot)$ and $\Gamma_2(\cdot)$), and send them through noise-free independent reporting channels to the FC. The FC uses the received data from both sensors and combine them using the fusion rule (i.e. AND, OR, XOR, or LRT) to generate the global decision. The received signal at Sensor j is z_j, and the purpose of this distributed detection system is to detect the existence of the PU (hypothesis H_1) on the

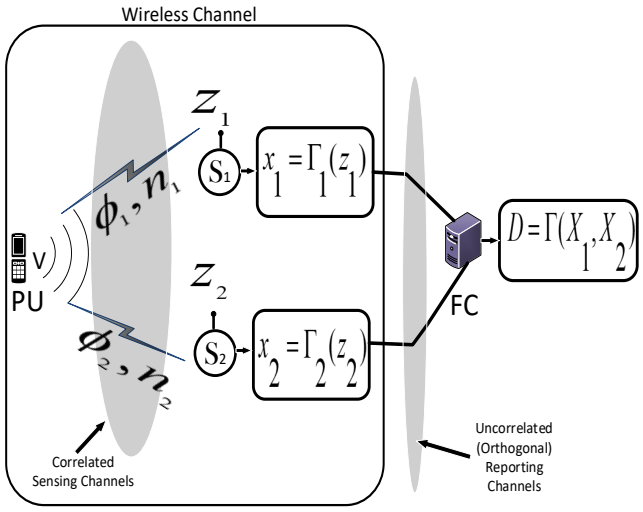


Figure 4.1. Distributed detection system.

given frequency channel, or its absence (hypothesis H_0). Under H_0, $z_j = n_j$, $j = 1, 2$; while under H_1, $z_j = \Phi_j V + n_j$, $j = 1, 2$. Where V is the PU transmitted signal level under H_1, and Φ_j and n_j are the attenuation and noise component on sensing channel j between the PU and Sensor j, $j = 1, 2$, respectively. After receiving z_j, Sensor j quantizes z_j into x_j which takes one of two levels: "0" or "1". The global decision at the FC (i.e. D) is also binary: "0" (i.e. H_0) and "1" (i.e. H_1). The received noise components at the sensors are assumed to be correlated with an unknown joint pdf, but by assuming the use of the copula method, the closest approximate joint pdf is assumed to be the Clayton copula joint pdf [42]. This means that z_1 and z_2 are also jointly correlated with Clayton copula joint pdf. Hence, we have the following

$$F_{Z_1,Z_2}(z_1, z_2) = C(u, v) = \left(u^{-\theta} + v^{-\theta} - 1\right)^{\frac{-1}{\theta}}\Big|_{u=F_{Z_1}(z_1),\ v=F_{Z_2}(z_2)}, \tag{4.1}$$

where $F_{Z_1,Z_2}(z_1, z_2)$ is the joint cumulative distribution function (CDF) of Z_1 and Z_2. $F_{Z_1}(z_1)$ and $F_{Z_2}(z_2)$ are the marginal CDFs of Z_1 and Z_2, respectively. In this work, we assume the full knowledge about $F_{Z_1}(z_1)$ and $F_{Z_2}(z_2)$. θ is the Clayton copula model parameter.

To study the effect of θ on the correlation between Z_1 and Z_2, we need to generate u

and v samples from the copula function in Equation (4.1) for a given θ, then find the corresponding z_1 and z_2 values. After generating an adequate number of samples, the correlation ($\hat{\rho}$) between the generated values of z_1 and z_2 can be calculated using the following relation

$$\hat{\rho} = \frac{\frac{1}{N}\sum_{i=1}^{N}(u_i - \overline{u})(v_i - \overline{v})}{\overline{\sigma}_1 \overline{\sigma}_2}, \tag{4.2}$$

where $\overline{u}$, $\overline{v}$, $\overline{\sigma}_1$, and $\overline{\sigma}_2$ are the mean value of the u samples, the mean value of the v samples, the standard deviation of the u samples, and the standard deviation of the v samples, respectively. To generate pairs of (u, v) according to $C(u, v)$, it can be shown that

$$\begin{aligned} w &= P(Z_2 \leq z_2 | Z_1 = z_1) \\ &= \frac{\partial C(u,v)}{\partial u} \\ &= u^{-(\theta+1)}(u^{-\theta} + v^{-\theta} - 1)^{-(\frac{1+\theta}{\theta})}. \end{aligned} \tag{4.3}$$

According to Equation (4.3), we have

$$v = \left((w^{\frac{-\theta}{1+\theta}} - 1)u^{-\theta} + 1 \right)^{\frac{-1}{\theta}}. \tag{4.4}$$

By randomly generating N pairs of (w, u), the corresponding v value can be found using (4.4). Note that $w, u \in [0, 1]$ and are generated randomly and independently from each other. This provides N samples of (u, v), where u and v are correlated according to (4.1). Hence, the corresponding (z_1, z_2) pairs can be generated according to $(z_1, z_2) = (F_{Z_1}^{-1}(u), F_{Z_2}^{-1}(v))$.

Figure 4.2 provides the relation between θ and $\hat{\rho}$. According to this figure, as θ increases, the correlation $\hat{\rho}$ approaches 1 in a one-to-one relation. We also calculated ρ theoretically using (4.1) (i.e the joint pdf of U and V) and the relation $\rho = \frac{\mathrm{Cov}(U,V)}{\sigma_1 \sigma_2}$. The results from this agree with the estimated ρ (i.e. $\hat{\rho}$) for large N (e.g. $N = 10000$).

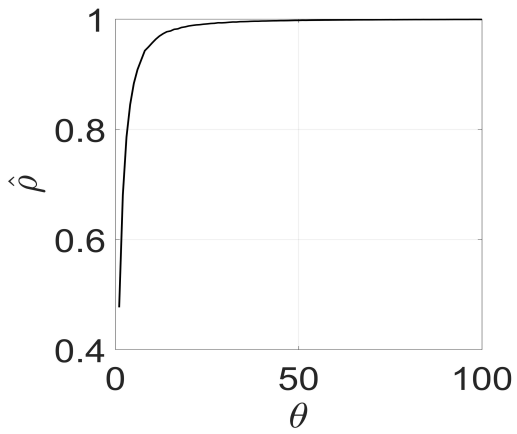


Figure 4.2. $\hat{\rho}$ Vs θ.

The local sensor decisions can be formulated in a general form as:

$$X_j = \Gamma_j(Z_j) = \begin{cases} 1, & \text{if } Z_j \in A_j \\ & \\ 0, & \text{if } Z_j \in \overline{A}_j \end{cases}, \forall j = 1, 2, \tag{4.5}$$

where A_j is the interval in which Sensor j decides "1" and $\overline{A}_j$ is the complement of A_j where Sensor j decides "0". It is worth mentioning that A_j and $\overline{A}_j$ are, in general, non-contiguous intervals unless we force them to be. This means that the local decision rules may have multiple thresholds separating the non-contiguous sub-intervals of A_j from the non-contiguous sub-intervals of $\overline{A}_j$.

In this chapter, we are interested in studying the P_e system performance in the following cases:

- Case 1: AND fusion rule and local sensor decision rules restricted by a single threshold.
- Case 2: OR fusion rule and local sensor decision rules restricted by a single threshold.
- Case 3: XOR fusion rule and local sensor decision rules restricted by a single threshold.
- Case 4: AND fusion rule and local sensor decision rules unrestricted by a single threshold.

- Case 5: OR fusion rule and local sensor decision rules unrestricted by a single threshold.
- Case 6: XOR fusion rule and local sensor decision rules unrestricted by a single threshold.
- Case 7: LRT fusion rule and local sensor decision rules unrestricted by a single threshold.

In the first three cases, the Exhaustive Search is used to find the decision thresholds at the sensors. In the rest of the cases, the GA-based method in [45, 48] is used to find the local sensor decisions.

It is assumed that the marginal pdfs of Z_1 and Z_2 are Gaussian while their joint pdf is Clayton copula. According to this, the problem formulations can be derived as follows.

Under H_0: $Z_1 \sim N(0, \sigma^2)$, $Z_2 \sim N(0, \sigma^2)$

$$\begin{aligned} f_{Z_1,Z_2|H_0}(z_1, z_2) &= \frac{\partial^2 C(u,v)}{\partial u \partial v} \frac{\partial u}{\partial z_1} \frac{\partial v}{\partial z_2} \\ &= \frac{\partial^2 C(u,v)}{\partial u \partial v} f_{Z_1|H_0}(z_1) f_{Z_2|H_0}(z_2), \end{aligned} \tag{4.6}$$

where $\frac{\partial^2 C(u,v)}{\partial u \partial v} = (\theta+1)(uv)^{(-\theta+1)}(u^{-\theta}+v^{-\theta}-1)^{(\frac{-(2\theta+1)}{\theta})}$, $u = F_{Z_1|H0}(z_1)$, and $v = F_{Z_2|H0}(z_2)$.

Under H_1: $Z_1 \sim N(\mu_1, \sigma^2)$, $Z_2 \sim N(\mu_2, \sigma^2)$

$$\begin{aligned} f_{Z_1,Z_2|H_1}(z_1, z_2) &= \frac{\partial^2 C(u,v)}{\partial u \partial v} \frac{\partial u}{\partial z_1} \frac{\partial v}{\partial z_2} \\ &= \frac{\partial^2 C(u,v)}{\partial u \partial v} f_{Z_1|H_1}(z_1) f_{Z_2|H_1}(z_2), \end{aligned} \tag{4.7}$$

where $\frac{\partial^2 C(u,v)}{\partial u \partial v} = (\theta+1)(uv)^{(-\theta+1)}(u^{-\theta}+v^{-\theta}-1)^{(\frac{-(2\theta+1)}{\theta})}$, $u = F_{Z_1|H1}(z_1)$, and $v = F_{Z_2|H1}(z_2)$.

In the system model presented in this work, we first divide the sensing range of Sensor j (i.e. form $-\infty$ to ∞) into N_j subareas. This can be achieved by dividing the sensing range

j into intervals with equal probability mass $\zeta_j = \frac{1}{N_j}$, under the marginal pdfs of Sensor j (i.e. $P(H_1)f_{Z_j|H_1}(z_j) + P(H_0)f_{Z_j|H_0}(z_j)$), where $P(H_1)$ and $P(H_0)$ are the prior probabilities of H_1 and H_0, respectively. Typically, we choose N_j to be large enough to compute significant probability mass variations. For Sensor j, assume that Subarea i is represented by the symbol $I_{i_j}^{(j)}, \forall i_j = 1, \ldots, N_j; j = 1, 2$. Let the set of variables corresponding to the subareas of Sensor j be $\{\alpha_1^{(j)}, \ldots, \alpha_{N_j}^{(j)}\}$, where $\alpha_i^{(j)} \in \{0, 1\}, \forall i = 1, \ldots, N_j; j = 1, \ldots, M$. For example, if $\alpha_3^{(2)} = 1$, it means when the received signal by Sensor 2 (i.e. z_2) falls in subarea 3, then it is quantized to $x_2 = 1$.

Assume that the subareas' thresholds of Sensor j on axis j are given by the vector $E_j = [e_1^{(j)} = -\infty, e_2^{(j)}, e_3^{(j)}, \ldots, e_{N_j+1}^{(j)} = \infty], \forall j = 1, 2$. Then, E_j's elements can be determined according to the equal-probability division procedure. Define $\underline{I}$ to be the subspace that results form the intersection of subareas $I_{i_1}^{(1)}$, and $I_{i_2}^{(2)}$. Hence, the probability of a given sub-space can be found as follows

$$P_{\underline{I}|H_k} = P(\underline{I}|H_k) = \int_{e_{i_1}^{(1)}}^{e_{i_1+1}^{(1)}} \int_{e_{i_2}^{(2)}}^{e_{i_2+1}^{(2)}} f_{Z_1,Z_2}(z_1, z_2) dz_1 dz_2, \forall k = \{0, 1\}.$$

Hence, we have the following problem formulations.

Case 1: AND fusion rule and local sensor decision rules restricted by a single threshold.

$$\min_{t_1,t_2} \Big(P_e = P(H_0)(C(1,1) + C(u^*, v^*) - C(u^*, 1) - C(1, v^*)) + P(H_1)(1 - C(1,1) - C(u^{**}, v^{**}) + C(u^{**}, 1) + C(1, v^{**})) \Big), \tag{4.8}$$

Case 2: OR fusion rule and local sensor decision rules restricted by a single threshold.

$$\min_{t_1,t_2}\Big(P_e = P(H_0)(1 - C(u^*, v^*)) + P(H_1)C(u^{**}, v^{**})\Big), \tag{4.9}$$

Case 3: XOR fusion rule and local sensor decision rules restricted by a single threshold.

$$\min_{t_1,t_2}\Big(P_e = P(H_0)(1 + 2C(u^*, v^*) - C(u^*, 1) - C(1, v^*)) + P(H_1)(-2C(u^{**}, v^{**}) + C(u^{**}, 1) + C(1, v^{**}))\Big), \tag{4.10}$$

where, in Cases 1, 2, and 3, $u^* = F_{Z_1|H_0}(t_1)$, $v^* = F_{Z_2|H_0}(t_2)$, $u^{**} = F_{Z_1|H_1}(t_1)$, $v^{**} = F_{Z_2|H_1}(t_2)$, $t_1 \in E_1$, $t_2 \in E_2$, and

$$X_j = \Gamma_j(Z_j) = \begin{cases} 1, & \text{if } Z_j \in A_j = [t_j, \infty) \\ 0, & \text{if } x_j \in \overline{A}_j = (-\infty, t_j) \end{cases}, \forall j = 1, 2.$$

Cases 4-7:

$$\min_{\alpha_{i_j}^{(j)}}\Big(P_e = P(H_1) + \sum_{\underline{x}}\sum_{\underline{i}} \delta \prod P(D = 1|x_1, x_2)\Big) \tag{4.11}$$

s.t.

$$\alpha_{i_j}^{(j)} \in \{0, 1\}, \forall i_j = 1, \ldots, N_j; j = 1, 2,$$

$$\delta = P(H_0)P_{\underline{I}|H_0} - P(H_1)P_{\underline{I}|H_1},$$

$$\prod = \psi(x_1 - \alpha_{i_1}^{(1)})\psi(x_2 - \alpha_{i_2}^{(2)}),$$

$$\psi(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\sum_{\underline{x}} = \sum_{x_1=0}^{1}\sum_{x_2=0}^{1} \text{ and } \sum_{\underline{i}} = \sum_{i_1=1}^{N_1}\sum_{i_2=1}^{N_2}.$$

- **If the fusion rule is AND (i.e. Case 4):**

$$P(D=1|x_1,x_2)=\begin{cases}1, & \text{if } (x_1,x_2)=(1,1)\\ 0, & \text{otherwise}\end{cases}.$$

- **If the fusion rule is OR (i.e. Case 5):**

$$P(D=1|x_1,x_2)=\begin{cases}0, & \text{if } (x_1,x_2)=(0,0)\\ 1, & \text{otherwise}\end{cases}.$$

- **If the fusion rule is XOR (i.e. Case 6):**

$$P(D=1|x_1,x_2)=\begin{cases}0, & \text{if } x_1=x_2\\ 1, & \text{otherwise}\end{cases}.$$

- **If the fusion rule is LRT (i.e. Case 7):**

$$P(D=1|x_1,x_2)=\begin{cases}1, & \text{if } \sum_{\underline{i}}(P_{\underline{I}|H_1}-\lambda P_{\underline{I}|H_0})\prod\geq 0\\ 0, & \text{otherwise}\end{cases},$$

where $\lambda=\frac{P(H_0)}{P(H_1)}$.

4.3 Optimization Algorithms

To find the solutions of the problem formulations presented in Section 4.2, the following two algorithms are used.

4.3.1 Exhaustive Search Algorithm (ESA)

This algorithm is used in Cases 1, 2, and 3. ESA has the following steps.

Step 1: Find all the combinations of the pair (t_1,t_2), where $t_1\in E_1$ and $t_2\in E_2$.

Step 2: Find the P_e value associated with each (t_1, t_2) combination from Step 1.

Step 3: Choose the (t_1, t_2) combination with the least P_e value from Step 2.

The time complexity of ESA, is approximately $2 \times |E_1| \times |E_2| = 2 \times (N_1+1)(N_2+1) \approx O(N_1 \times N_2)$. Note that Step 3 has low complexity compared to Steps 1 and 2.

4.3.2 Genetic Algorithm (GA)

This algorithm is used in Cases 4-7. The GA has the following steps.

Step 1: Generate a set $\mathcal{G}$ of solutions (i.e. a generation/set of chromosomes). Let the number of solutions in this set be $|\mathcal{G}|$ and generate these solutions by randomly assigning values to the "α"s in (4.11).

Step 2: Evaluate the fitness function for each solution in $\mathcal{G}$ (i.e. $1 - P_e$ of each solution).

Step 3: Take a number of solutions that have the highest fitness function values from $\mathcal{G}$ and directly place them into the next generation of $\mathcal{G}$. In this work, the chosen number of solutions is $\frac{|\mathcal{G}|}{2}$.

Step 4: Choose $\frac{|\mathcal{G}|}{4}$ pairs from the current $\mathcal{G}$ according to their fitness value by using the roulette wheel selection method. Then perform c number of random crossovers between each pair to generate two new solutions. Doing this for all $\frac{|\mathcal{G}|}{4}$ chosen pairs produces $\frac{|\mathcal{G}|}{2}$ new solutions. These new solutions are used to fill the second half of the next generation (i.e. the $\mathcal{G}$ of the next iteration).

Step 5: Perform mutation on the new solutions resulted from Step 4 by flipping each bit with probability $\frac{\epsilon}{100}$.

Step 6: Repeat steps 2 to 5 for $Iter$ number of iterations.

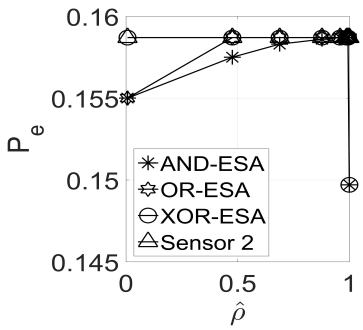
Step 8: Take the highest fitness valued solution from the resulting $\mathcal{G}$ as the final solution of (4.11).

For the time and space complexity of GA, refer to [45, 48].

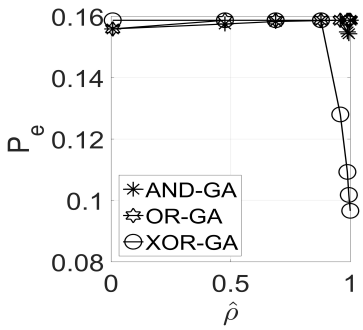
4.4 Performance Evaluation

In these simulations, the system shown in Figure 4.1 is adopted and the Clayton copula joint pdf, in Equations (4.6) and (4.7), is used to model the unknown joint pdf of Z_1 and Z_2, under H_0 and H_1, respectively. The marginal pdfs of Z_1 and Z_2 are assumed to be known Gaussian pdfs under both H_0 and H_1. In [46] and [48], the authors showed that the correlation can be used advantageously to get lower P_e at the FC. While in [46] the authors proved this to be true in [48] for Gaussian joint pdfs in centralized detection system. They also proved it to be true for any type of joint pdf in a decentralized (distributed) detection system with strong correlation (i.e. $\rho \to 1$) if and only if 1) the sensors are allowed to have multiple number of decision thresholds, to separate the subareas of A (i.e. H_1 decision interval) and $\overline{A}$ (i.e. H_0 decision interval), 2) the FC center uses the optimal fusion rule, and 3) the received signals at the sensors (i.e. Z_1 and Z_2) have different mean values. It is important to test the proposed system in this paper against these rules as in the following simulations to test their ability in capturing the correlation and utilizing it to minimize the P_e at the FC.

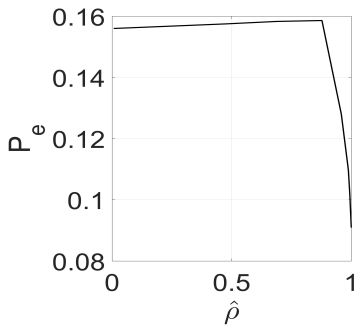
Simulations 1: In these simulations, we are interested in observing the P_e behavior with respect to changes in θ values. Assume under H_0: $Z_1 \sim N(0, \sigma)$, $Z_2 \sim N(0, \sigma)$, and Z_1 and Z_2 are jointly distributed according to the Clayton copula in Equation (4.6). Under H_1: $Z_1 \sim N(\mu_1, \sigma)$, $Z_2 \sim N(\mu_2, \sigma)$, and Z_1 and Z_2 are jointly distributed according to the Clayton copula in Equation (4.7). $\sigma = 1$, $\mu_1 = 1$, $\mu_2 = 2$, and $P(H_0) = P(H_1) = 0.5$. The GA parameters are: $N_1 = N_2 = 10$, $Iter = 6000$, $c = 2$, $|\mathcal{G}| = 6$, and $\epsilon\% = 5\%$. Regarding the ESA, $N_1 = N_2 = 100$. Due to the fact, that the correlation $\hat{\rho}$ changes slowly with θ for large values of θ, it is hard to see the changes in P_e if it is plotted against θ. Hence, here we plot P_e Vs $\hat{\rho}$ which makes understanding the figure easier by looking at the range of $|\hat{\rho}| \in (0, 1)$ instead of $\theta \in (0, \infty)$. Table 4.1 provides the $\hat{\rho}$ value for each given value of θ using Equation (4.2) and $N = 10000$ samples.



(a) Cases 1, 2, 3, and Sensor 2.

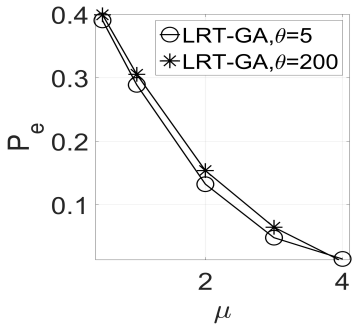


(b) Cases 4, 5, and 6.

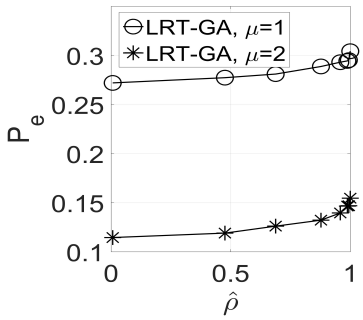


(c) Case 7.

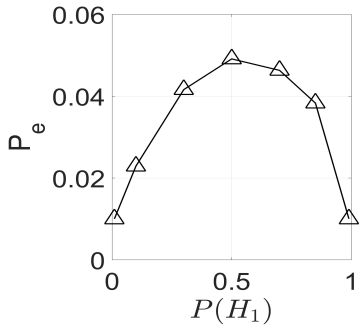
Figure 4.3. P_e Vs $\hat{\rho}$



(a) P_e Vs μ.



(b) P_e Vs $\hat{\rho}$.



(c) P_e Vs $P(H_1)$.

Figure 4.4. P_e Vs μ, $\hat{\rho}$, and $P(H_1)$.

θ	0.0001	1	2	5	10	20	30	200
$\hat{\rho}$	0.007	0.4762	0.6881	0.8773	0.9582	0.9876	0.9937	0.9998

Table 4.1. θ Vs $\hat{\rho}$

In Figure 4.3(a), Case 1 is represented by the curve AND-ESA, Case 2 is represented by the curve OR-ESA, and Case 3 is represented by the curve XOR-ESA. In Figure 4.3(b), Case 4 is represented by the curve AND-GA, Case 5 is represented by the curve OR-GA, Case 6 is represented by the curve XOR-GA, and the P_e curve resulting from the use of Sensor 2 only for detection is represented by the curve "Sensor 2" (see [65] for more about single sensor detection). Lastly, the curve in Figure 4.3(c) represents Case 7 (i.e. LRT-GA). According to these figures, all the cases are not able to provide the best P_e performance at all $\hat{\rho}$ values except for LRT-GA. In Figure 4.3(a), the betterment of the P_e is not the best at all $\hat{\rho}$ values for all the given cases. The performance of Sensor 2 (i.e. if the global decision

is made by Sensor 2 only) is used as a reference line to judge the system performance (note that the performance of Sensor 1 only is even worse with a $P_e = 0.3085$). In 4.3(b), it is also obvious that the XOR-GA case/AND-GA and OR-GA cases fail to provide low P_e values at low/high $\hat{\rho}$. The AND-ESA, OR-ESA, and XOR-ESA behavior is expected since they do not satisfy all the conditions provided in [48] to gain advantage of the correlation. These cases 1) do not allow multiple thresholds at the sensors, 2) do not use the optimal fusion rule at the FC. Regarding AND-GA and OR-GA, even though they satisfy the condition of allowing multiple thresholds but they do not use the optimal fusion rules which resulted in a poor P_e performance as $\hat{\rho}$ increases. Regarding XOR-GA, it provides significant decrease in P_e and $\hat{\rho} \to 1$. This is expected since according to [48], the LRT at the FC converges to XOR as $\rho \to 1$ while having different mean signals at the sensors. Hence, this case (i.e. Case 6) satisfies all the conditions provided in [48] at $\hat{\rho} \to 1$. On the other hand, it fails at small $\hat{\rho}$ values, since XOR is not the optimal fusion rule at these values. Lastly, LRT-GA in Figure 4.3(c) is able to gain advantage from the correlation and use the optimal fusion rule at any given $\hat{\rho}$ value. As a result of satisfying all the needed conditions.

In the adopted system model, the fusion rule in Case 7 (i.e. LRT) should converge to one of four possible forms at any given $\hat{\rho}$, namely: AND, OR, XOR, or ignore one sensor [76]. In these simulations, Table 4.2 shows the fusion rule that the LRT converged to for the given $\hat{\rho}$ value in Figure 4.3(c).

$\hat{\rho}$	0.007	0.4762	0.6881	0.8773	0.9582	0.9876	0.9937	0.9998
LRT	$X_1\overline{X}_2$	$\overline{X}_1\overline{X}_2$	$X_1\overline{X}_2$	$\overline{X}_1\overline{X}_2$	XOR	XNOR	XNOR	XOR

Table 4.2. Case 7 fusion Vs $\hat{\rho}$

Note that any fusion rule in Table 4.2 is just a modified version of one of four rules: AND, OR, XOR, and ignore one sensor. For example, the rule $\overline{X}_1\overline{X}_2$ is another form of an AND rule, since we can invert the resulting local decision rules from the GA at both sensors (i.e. switch the resulting decision regions at each senor). By doing this, the fusion rule becomes X_1X_2 which is an AND rule. It is also straight forward to show that the XNOR

(i.e. the compliment of XOR) rule can be converted to XOR rule by making the appropriate changes. Based on these results, we can see that the Cases 1-6 do not always provide the best P_e performance. Hence, the simulations that follow consider the system behavior in Case 7 only.

Simulations 2: In these simulations, the P_e behavior is studied as the received signal mean μ increases. The system has the same properties as in Simulations 1 except for the values of μ_1 and μ_2, where $\mu_1 = \mu_2 = \mu$, and $\theta = \{5, 200\}$ which corresponds to $\hat{\rho} = \{0.8773, 0.9998\}$.

As Figure 4.4(a), the P_e value decreases as the received power at the sensors increases. This is expected since increasing the signal levels increases the distance between the joint pdfs of H_0 and H_1. Hence, the overlapping space between them decreases which results in more accurate decisions and a reduction in the overall P_e value. We also notice that the curve corresponding to $\theta = 200$ provide higher P_e values than the curve of $\theta = 5$, this behavior has been observed in similar simulations provided in [46] and [48] in which the joint Gaussian pdf was used. This behavior is observed when the received signals have the same mean, hence, as $\hat{\rho}$ increases, the P_e value increases. This behavior was explained for the centralized detection for the joint Gaussian pdf in [46]. It is also explained for any joint pdf in distributed detection systems as $\rho \to 1$ in [48]. Figure 4.4(b) also provide the same P_e behavior using the Clayton pdf with two different μ values.

Simulations 3: In these simulations, the P_e behavior is studied as $P(H_1)$ increases. The system has the same properties as in Simulations 1 except for the values of μ_1 and μ_2, where $\mu_1 = \mu_2 = 3$, and $\theta = 2$ which corresponds to $\hat{\rho} = 0.6881$.

As it is obvious from the figure, the P_e has its highest value at $P(H_1) = 0.5$ which is expected since at this prior probability the most uncertainty, on which hypothesis is true, happens [46, 45, 48].

Simulations 4: In these simulations, we present graphically the fusion rule decision areas in a scenario of discontiguous local decision areas. Assuming the same system setup

as in Simulations 1 and choosing $\theta = 200$ and fusion rule to be LRT (i.e. Case 7). With this system setup, the LRT fusion rule converges to XOR fusion rule. The resulting local decisions and the XOR fusion rule from the GA are as follows.

$$X_1 = \Gamma(Z_1) = \begin{cases} 1, & \text{if } Z_1 \in (-\infty, 0.213) \cup (1.9394, \infty) \\ 0, & \text{if } x_1 \in (0.213, 1.9394) \end{cases}, \tag{4.12}$$

$$X_2 = \Gamma(Z_2) = \begin{cases} 1, & \text{if } Z_2(-\infty, 0.1676) \cup (1.4135, 2.8495) \\ 0, & \text{if } x_2 \in (0.1676, 1.4135) \cup (2.8495, \infty) \end{cases}, \tag{4.13}$$

$$D(X_1, X_2) = \begin{cases} 0, & \text{if } X_1 = X_2 \\ 1, & \text{otherwise} \end{cases} = \text{XOR}(X_1, X_2). \tag{4.14}$$

All these rules can be visualized together in Figure 4.5. The Z_1-axis and Z_2-axis represent Sensor 1's and Sensor 2's measurements, respectively. The dark gray spots on the figure are where the FC decides H_0 to be true (i.e. $D =$"0"), while the light gray spots are where the FC decides H_1 to be true (i.e. $D =$"1").

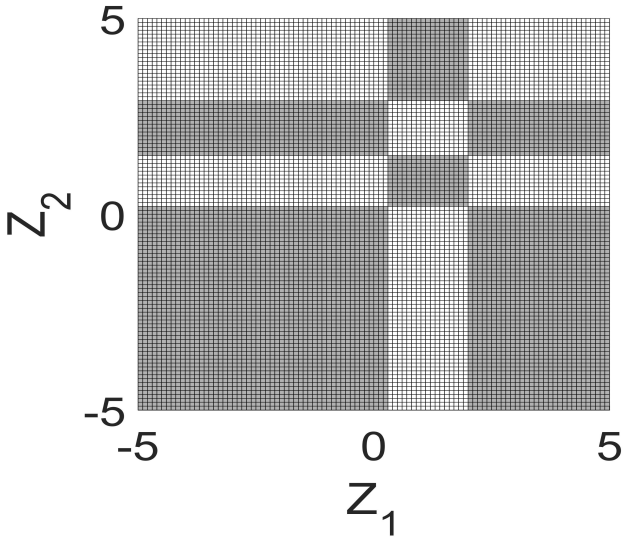


Figure 4.5. Local decision and fusion rules visualization.

4.5 Conclusion

In this chapter the probability of error behavior of two-sensor distributed detection system is studied. Several boolean and likelihood ratio test fusion rules are tested assuming a Clayton copula model. The results show the failure of the boolean functions to achieve the best performance at any given system setup, while the likelihood ratio test is the only test that converged to the best fusion rule and achieved the best probability of error performance at any given system setup. The aim of this chapter is to study Clayton distribution in distributed detection system and observe the performance in terms of correlation, signal power, and prior probabilities changes. As our results show, the Clayton distribution has behavior similar to the Gaussian distribution considered in earlier chapters.

CHAPTER 5

Generous Throughput Oriented MAC Protocol for Infra-structured WiFi Networks

Due to the random channel assignment, the IEEE 802.11 medium access control (MAC) protocol often suffers from severe network throughput degradation. Although some proposals in the literature tried to solve this problem, most of them are either of high complexity or based on some rigid or unrealistic assumptions. This chapter presents an efficient throughput oriented protocol, which is characterized by its simplicity and compliance to infra-structured IEEE 802.11 networks. The core idea of this novel MAC protocol comes from its channel assignment mechanism, which chooses the frequency channel that is just good enough to maximize the number of served users per access point (AP) and leaves the over-qualified channels to other APs that may be in need for using them. Unlike other proposed methods which aimed to solve the problem while ignoring some important network factors (e.g. out-of-system interference, the random AP deployment in reality, APs being owned by multiple networks, etc.), the proposed MAC protocol takes these factors into consideration. The proposed MAC protocol is compared to the Random, the Greedy, and the CACAO[1] channel assignment based IEEE 802.11 MAC protocols. The results show a remarkable throughput and fairness improvement of the proposed protocol while reserving simplicity and realism.

5.1 Introduction

5.1.1 Motivation

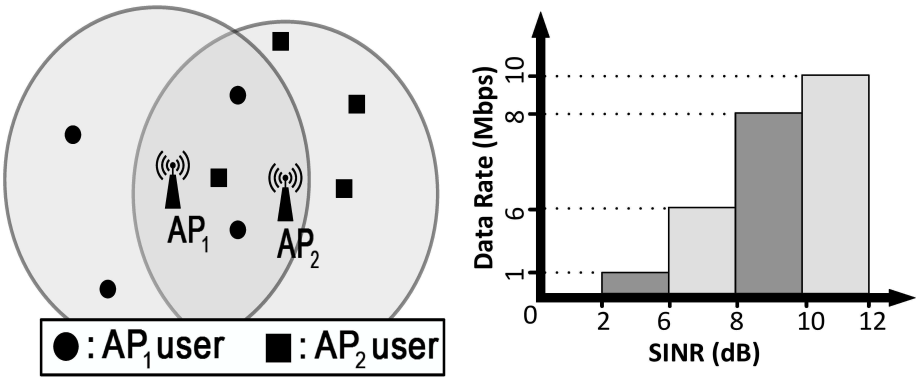
Due to the convenience installation and use, WiFi networks have gained a huge popularity. The main reason behind this is the use of the free open ISM (Industrial, Scientific, and

1 CACAO: Distributed client assisted channel assignment optimization for uncoordinated WLANs, [80].

Medical) frequency band for communication. The two main problems of this band are being a very confined band and suffering from the interference coming from other systems that use the same band (e.g. cordless phones, Bluetooth, microwave ovens, etc.) [56]. Because of these problems and the fact of the exponential increase of the WiFi network users [12], an efficient WiFi MAC protocol with wise frequency channel assignment[1] mechanism is thus demanded in order to maximize the network throughput and accommodate as many network users as possible. However, in the current IEEE 802.11 protocol series, there is no consideration for these problems as the assignment for the frequency channels per AP is totally random, [1, 2, 3], (i.e., Random channel assignment mechanism), which can significantly reduce the network throughput performance. On the other hand, the literature contains a number of WiFi MAC protocols that attempted to solve the randomness caused problem in the IEEE 802.11 channel assignment. The authors in [62] attempted to solve the problem by choosing the channel with the least co-channel interference. This is often referred to as a Greedy channel assignment, which may lead to poor performance in terms of fairness and aggregated network throughput. The work in [81] and [33] solved the problem under the assumption of full control on APs' positions. This assumption, however, contradicts the random deployment nature of the WiFi networks in general. In [81, 33, 55, 35, 14], the authors proposed a solution that assumes active collaborations between the APs. This assumption may not always hold in reality, since the APs are often owned by different administrators. This solution also requires high overhead. In addition, the work in [6, 81, 33], and [80] ignores the out-of-system interference. Such interference can degrade the network performance immensely. Hence, any reliable solution should take such factors into consideration.

This chapter proposes a new channel assignment mechanism called "Generous Throughput Oriented Channel Assignment" (GTCA) that can outperform both the Random and the Greedy channel assignments in terms of throughput. We illustrate the core idea behind the proposed GTCA channel assignment and its potential by a simple example given in Fig.

1 In this chapter, the terms of channel assignment and frequency channel assignments are used interchangeably.



(a) WiFi System Model. (b) SINR-Data rate relation.
Figure 5.1. WiFi model example and its SINR-Data rate relation.

5.1(a), which consists of two APs with two channels. This example is used to demonstrate the throughput performance of the Random, the Greedy, and the GTCA channel assignment mechanisms.

1) Random channel assignment: Fig. 5.1(a) reveals the inefficiency of this channel assignment in terms of maximizing the network throughput. An obvious scenario is to have the two APs in the same vicinity while operating on the same channel. Hence, the two APs totally interfere with each other and can not transmit simultaneously. This results in a huge reduction in the network throughput, which will be shown in Section 5.5.

2) Greedy channel assignment: This channel assignment chooses the channel that has the highest signal-to-interference-and-noise ratio (SINR) among all the channels supported by the system. Since the relation of the SINR and data rate is a staircase function in IEEE 802.11, both APs (in Fig. 5.1(a)) use the staircase SINR-Data rate relation shown in Fig. 5.1(b).

If there are only two channels (channel 1 and channel 2) for the APs to choose from and AP_1 (AP_2) senses 7dB (6.5dB) SINR on channel 1 and 3dB (4dB) on channel 2; then if AP_1 wants to transmit a *pdf* file that needs roughly 1Mbps data rate and AP_2 needs to transmit a high definition (HD) video that needs roughly 5Mbps data rate, then by using the Greedy channel assignment, if AP_1 gets its transmission first, it will choose channel 1 for its transmission, since it has the highest SINR. According to Fig. 5.1(b), when the SINR is

7dB, then this provides a 6Mbps data rate which is more than enough for AP_1 that needs just 1Mbps. Based on this, if AP_2 needs to transmit its HD video, the only channel option left is channel 2. Since the SINR sensed on channel 2 by AP_2 is 4dB, then the supported data rate is only 1Mbps. This data rate is not sufficient for AP_2 to complete its transmission. Therefore, the total throughput in this example is 1Mbps and the total blocked throughput is 5Mbps. This means only 17% of the possible throughput for transmission is transmitted while the rest is blocked. Next we show how the GTCA makes a complete transmission for the possible throughput in this example.

3) Generous throughput oriented channel assignment (GTCA): The basic idea behind this channel assignment is to let the AP choose the channel that has the lowest SINR but meets the needed rate. For clarity, let us look at the same aforementioned example of the Greedy channel assignment. According to GTCA, if AP_1 gets the transmission request before AP_2, then it chooses the channel that has the smallest SINR but satisfies its needed rate, which is channel 2 that has 3dB SINR. The rate supported by channel 2 is 1Mbps, which is enough for AP_1 that needs 1Mbps. If AP_2 wants to transmit, then it can use channel 1, since it is not being used by AP_1. Because the rate supported by channel 1 is 6Mbps (since AP_2 can sense 6.5dB SINR on this channel), then it satisfies the needed rate by AP_2 (5Mbps). The total throughput here is (1+5)=6Mbps, and the total blocked throughput is 0Mbps. Therefore, using GTCA, the utilization of the possible throughput is 100% with an improvement equals to 83% compared to the Greedy channel assignment for this specific example. This example shows the reason behind the name "Generous", where the AP leaves the over-qualified channels for other APs that may be in need for higher data rates provided by these channels. As a result, this can increase the overall network throughput, which makes this channel assignment "Throughput oriented".

5.1.2 Related Work

In the literature, there are some works proposed to tackle the Random channel assignment problem of the IEEE 802.11 protocol. In [81], the authors used the Shannon capacity maximization to calculate the optimal number of APs in a given area (i.e., the density of APs) under the condition of worst case scenario when all APs have the same frequency channel. The main assumption there is the regular (i.e., not random) deployment of APs in the area. Then they derived a mixed integer linear programming (MILP) problem formulation to assign a channel for each AP in the network. The assigned channels are taken from a pool of partially overlapping channels while having the objective of minimizing the total network interference. The authors found their MILP problem to be an NP-complete problem. To solve this problem optimally, the authors considered the interference coming only from the closest three tiers of interfering APs. The authors provided a two-phase algorithm to solve this problem formulation. This algorithm first calculates the optimal number of APs in a given area while all APs have the same channel. Then this algorithm provides the optimal channel assignment for the given regular AP deployment.

In [33], the authors provided a solution for the channel assignment problem by maximizing the minimum distance between the APs. It was shown that the minimum distance happens when the APs operate on the same channel. Depending on this, maximizing the minimum distance guarantees maximizing the distance for any other combination of channel assignments and thus minimizing the inter-cell interference. The authors formulated their problem as an MILP problem and solved it using the free-ware optimization solver "LP-SOLVE" in [16].

In [68], the authors considered the problem of having out-of-system devices. The authors used their previously proposed idea of dynamic channel assignment (DCA) for the the APs. In this DCA, the APs make periodic measurements for the amount of interference they experience on all channels. Then each AP lists the channels in increasing order according to the corresponding interference value observed. After that, the AP simply chooses the

channel with the least level of interference. The authors of [68] used this idea of DCA with the modification that the AP uses the second least interference valued channel instead of the least one to address the problem of the increased number of users in the system.

In [55], the authors took the problem of inter-cell interference into consideration. Such a problem can severely reduce system efficiency. Hence, the authors proposed a centralized channel allocation scheme for the IEEE 802.11 WiFi networks. According to this, all the APs are managed by a centralized controller (CC) through a control channel. The APs scan the system's channels and measure the utilization on each one of them (i.e., the number of APs operating on each channel). Then each AP sends its measurement to the CC to make the channel assignment. The authors formulated this channel assignment as finding the maximum weight matching on bipartite graph. The Hungarian method [51] is used to solve such optimization problems.

5.1.3 Contribution and Comparison

This chapter proposes a new IEEE 802.11 based MAC protocol for infra-structured WiFi networks. Through the rest of this chapter, this MAC protocol is called "Generous Throughput Oriented MAC protocol" (GT-MAC), which takes the GTCA as its frequency channel assignment mechanism to address the problem of using the Random frequency channel assignment. The main attractive features of the GT-MAC and its GTCA, along with a comparison to some of the proposed solutions in the literature, are listed below.

- Dynamic and fully automated channel assignment with no need for human interaction, unlike the currently used in IEEE 802.11 [1, 2, 3, 4].
- Negligible channel assignment overhead compared to [81, 33, 35, 80]. In fact this overhead is, at worst, double that in the current IEEE 802.11 standard in terms of the transmitted control packets number.
- The GTCA is a generous (non-greedy) channel selection mechanism and throughput oriented, since it allows other APs to use the over-qualified channels for their transmissions,

unlike the greedy methods that use the best available channel for transmission (e.g. [62], [9]).

- No restrictions on the positions or density of the APs in a given area, which is not considered in some of the previously proposed works (e.g. [81, 33]).
- No need for active collaborations between the APs. Unlike [35], using passive learning is sufficient to fulfill the channel assignment process.
- No need for power control, since this assignment mechanism uses a fixed power level, as in the IEEE 802.11 standards.
- Highly compatible with the IEEE 802.11. GT-MAC can be obtained with only small changes on the existing IEEE 802.11 standards.
- No need for the APs to be owned by the same administrator (unlike [81, 33, 55, 14]). The possibility of having different administrators can still be valid, and the GT-MAC can fully operate. This is due to the lack of the assumption on the direct connectivity between the APs.
- The GT-MAC also takes into account the out-of-system interference, [14], which is not considered in some of the previously proposed works (e.g. [6, 81, 33, 80]).
- High level of interference protection for the first and second tier neighboring APs.
- GTCA is formulated as an optimization problem, and a solution algorithm is provided that can solve this optimization problem optimally in linear time.

5.2 Network Model and Problem Statment

5.2.1 WiFi General System Model

The current WiFi systems are very similar to the demonstration in Fig. 5.1(a), which consists of a number of different APs. Each one of them represents a cell. Within each cell

there are multiple users that can communicate with each other on a MAC (medium access protocol) layer level through the AP. In this system there is no communication between the APs on the MAC layer level, which can escalate the problem of inter-cell interference, [55] (i.e., the interference between different cells). Accordingly, the current 802.11 MAC protocols should be modified to address this issue. In a specific geographical area, we consider a WiFi network that consists of N randomly deployed APs, given in the set $\boldsymbol{\mathcal{AP}} = \{AP_i; i = 1, \ldots, N\}$ that coexist with M out-of-system devices (OSDs), given in the set $\boldsymbol{\mathcal{OSD}} = \{OSD_m; m = 1, \ldots, M\}$. The OSDs are devices that use the same unlicensed band for other applications than WiFi. Hence, it is very important to consider the impact of these randomly deployed devices on the network performance. Each AP_i has a set of randomly distributed active users (AUs) within a circle of the radius r_{Max} around AP_i. These AUs of AP_i are given by the set $\boldsymbol{\mathcal{AU}}_i = \{AU_{j,i}; j = 1, \ldots, K_i\}$,$\forall i \in \{1, \ldots, N\}$, where K_i is the total number of the AUs that need to be served by AP_i. Note that $K_i \leq K_{Max}, \forall i \in \{1, \ldots, N\}$, where K_{Max} is the maximum number of AUs that can be supported by any AP in the system simultaneously. The set of the frequency channels that can be used by the system is defined as $\boldsymbol{\mathcal{C}} = \{C^q; q = 1, \ldots, c\}$, where c is the total number of frequency channels in the system.

5.2.2 Problem Statement

It is assumed that each AP keeps sending Beacon packets to share its basic information (e.g. AP identification (ID), its current channel of operation and the first tier neighboring AP channel list), and to keep the connectivity with its users. It is also assumed that an AP and its AUs share their needed transmission information (e.g. the needed data rate, the targeted receiver ID and interference and noise (IN) level on each channel) through control packets (i.e., send request to send [RTS] packets and receive clear to send [CTS] packets) on the current channel of operation within a specified time access window (AW) that is divided into a specific number of time slots (TSs). Each TS is accessed by the AP or one of its AUs using the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) access

strategy. After the channel assignment, the AP sends one beacon packet on the current channel to all of its AUs to inform them about the new channel assignment and which of them can and can not transmit. A detailed explanation of the proposed GT-MAC protocol is presented in Section 5.4.

For the channel assignment, each AP_i listens to the neighboring (first tier) AP beacons and then divides the set $\boldsymbol{\mathcal{C}}$ into three mutually exclusive sets (i.e., $\boldsymbol{\mathcal{C}}_i^f, \boldsymbol{\mathcal{C}}_i^s$ and $\boldsymbol{\mathcal{C}}_i^n$). In particular, $\boldsymbol{\mathcal{C}}_i^f = \{C_i^{f,q}; q = 1, \ldots, c_i^f\}$ and $\boldsymbol{\mathcal{C}}_i^s = \{C_i^{s,q}; q = 1, \ldots, c_i^s\}$ are the sets of the first and second tier neighboring APs' channels of AP_i, respectively. Where c_i^f and c_i^s are the total number of channels in $\boldsymbol{\mathcal{C}}_i^f$ and $\boldsymbol{\mathcal{C}}_i^s$, respectively, at a given moment, $\forall i = 1, \ldots, N$. Note that the set $\boldsymbol{\mathcal{C}}_i^s$ may contain the channels that are in the sets $\boldsymbol{\mathcal{C}}_j^f;\ j \neq i,\ j = 1, \ldots, N$ but not in the set $\boldsymbol{\mathcal{C}}_i^f$, and AP_j must be a first tier neighbor to AP_i. The set $\boldsymbol{\mathcal{C}}_i^n = \{C_i^{n,q}; q = 1, \ldots, c_i^n\}$ contains the channels that are in $\boldsymbol{\mathcal{C}}$ but not in $\boldsymbol{\mathcal{C}}_i^f \cup \boldsymbol{\mathcal{C}}_i^s$, where c_i^n is the total number of channels in this set. Note that the superscripts f, s, and n refer to the words: *first*, *second*, and *neither*, respectively. At any given moment $c_i^f + c_i^s + c_i^n = c, \forall i = 1, \ldots, N$.

At a given moment, each AP_i senses the IN on each channel in the system and constructs three corresponding sets. These sets are $\boldsymbol{\mathcal{I}}_i^f = \{I_i^{f,q}; q = 1, \ldots, c_i^f\}, \boldsymbol{\mathcal{I}}_i^s = \{I_i^{s,q}; q = 1, \ldots, c_i^s\}$, and $\boldsymbol{\mathcal{I}}_i^n = \{I_i^{n,q}; q = 1, \ldots, c_i^n\}$, where these sets contain the corresponding IN value at AP_i on each channel in the sets $\boldsymbol{\mathcal{C}}_i^f, \boldsymbol{\mathcal{C}}_i^s$, and $\boldsymbol{\mathcal{C}}_i^n$, respectively, $\forall i = 1, \ldots, N$. On the other hand, the sets of the IN values at $AU_{j,i}$ on each channel in $\boldsymbol{\mathcal{C}}_i^f$,$\boldsymbol{\mathcal{C}}_i^s$, and $\boldsymbol{\mathcal{C}}_i^n$ are given by $\boldsymbol{\mathcal{I}}_{(j,i)}^f = \{\mathcal{I}_{j,i}^{f,q}; q = 1, \ldots, c_i^f\}$, $\boldsymbol{\mathcal{I}}_{(j,i)}^s = \{\mathcal{I}_{j,i}^{s,q}; q = 1, \ldots, c_i^s\}$, and $\boldsymbol{\mathcal{I}}_{(j,i)}^n = \{\mathcal{I}_{j,i}^{n,q}; q = 1, \ldots, c_i^n\}$, respectively, $\forall j = 1, \ldots, K_i; i = 1, \ldots, N$. Note that $AU_{j,i}$ does not actually know the sets $\boldsymbol{\mathcal{C}}_i^f$,$\boldsymbol{\mathcal{C}}_i^s$, and $\boldsymbol{\mathcal{C}}_i^n$. So that, it just senses the IN on all channels and send these values to AP_i (ordered according to their corresponding channel's indexes). Then the AP just categorize these IN values into the sets $\boldsymbol{\mathcal{I}}_{(j,i)}^f$, $\boldsymbol{\mathcal{I}}_{(j,i)}^s$, and $\boldsymbol{\mathcal{I}}_{(j,i)}^n$ according to their channel's indexes and its place in the sets $\boldsymbol{\mathcal{C}}_i^f$,$\boldsymbol{\mathcal{C}}_i^s$, and $\boldsymbol{\mathcal{C}}_i^n$. For simplicity, it can be said that $AU_{j,i}$ sends $\boldsymbol{\mathcal{I}}_{(j,i)}^f$, $\boldsymbol{\mathcal{I}}_{(j,i)}^s$, and $\boldsymbol{\mathcal{I}}_{(j,i)}^n$ to AP_i.

Each AU$-AP_i$ connection needs a specific data rate value to fulfill its transmission.

These data rate values of AP_i's AUs are given by the set $\boldsymbol{\mathcal{R}}_i = \{R_{j,i}; j = 1, \ldots, K_i\}, \forall i = 1, \ldots, N$. The adopted propagation model is the simplified path loss model, [37]. The average received signal power according to this propagation model is given by the relation

$$P_r(d) = P_t\Big(\frac{\lambda^2 G_t G_r}{(4\pi)^2 d_0^2}\Big)\Big(\frac{d_0}{d}\Big)^{\alpha}, \tag{5.1}$$

where $P_r(d)$ is the average received signal power at distance d from the transmitter, P_t is the transmitted power, λ is the wavelength of the used frequency, G_t is the transmitter gain, G_r is the receiver gain, d_0 is the reference distance that is between 1m to 10m for the indoor environment, and α is the pathloss exponent which is approximately 3.5 for the same floor indoor environment, [37].

Using the simplified pathloss model, the sets $\boldsymbol{\mathcal{P}}^f_{(j,i)} = \{P^{f,q}_{j,i}; q = 1, \ldots, c^f_i\}$, $\boldsymbol{\mathcal{P}}^s_{(j,i)} = \{P^{s,q}_{j,i}; q = 1, \ldots, c^s_i\}$, and $\boldsymbol{\mathcal{P}}^n_{(j,i)} = \{P^{n,q}_{j,i}; q = 1, \ldots, c^n_i\}$ are defined as the sets that represent the received power at AP_i (or $AU_{j,i}$) from $AU_{j,i}$ (or AP_i) on each channel from the sets $\boldsymbol{\mathcal{C}}^f_i, \boldsymbol{\mathcal{C}}^s_i$, and $\boldsymbol{\mathcal{C}}^n_i$, $\forall j = 1, \ldots, K_i, \forall i = 1, \ldots, N$.

The SINR at AP_i on the channels $C^{f,q}_i$, $C^{s,q}_i$, and $C^{n,q}_i$ from $AU_{j,i}$ can be calculated as $\frac{P^{f,q}_{j,i}}{I^{f,q}_i}$, $\frac{P^{s,q}_{j,i}}{I^{s,q}_i}$, and $\frac{P^{n,q}_{j,i}}{I^{n,q}_i}$, respectively. On the other hand, the SINR at $AU_{j,i}$ on the channels $C^{f,q}_i$, $C^{s,q}_i$, and $C^{n,q}_i$ from AP_i can be calculated as $\frac{P^{f,q}_{j,i}}{I^{f,q}_{j,i}}$, $\frac{P^{s,q}_{j,i}}{I^{s,q}_{j,i}}$, and $\frac{P^{n,q}_{j,i}}{I^{n,q}_{j,i}}$, respectively. By defining the SINR-Data rate relation, $Rate = f(S)$, where S is the SINR on a given frequency channel, and $Rate$ is the associated data rate value according to the function $f(\cdot)$ (e.g., the staircase SINR-Data rate relation shown in Fig. 5.1(b)). Using this SINR-Data rate relation, we can define the sets that represent the actual data rate supported by the channels $C^{f,q}_i$, $C^{s,q}_i$, and $C^{n,q}_i$ between $AU_{j,i}$ and AP_i as $\boldsymbol{\mathcal{R}}^f_{(j,i)} = \{R^{f,q}_{j,i} = f\big(\min(\frac{P^{f,q}_{j,i}}{I^{f,q}_i}, \frac{P^{f,q}_{j,i}}{I^{f,q}_{j,i}})\big); q = 1, \ldots, c^f_i\}$, $\boldsymbol{\mathcal{R}}^s_{(j,i)} = \{R^{s,q}_{j,i} = f\big(\min(\frac{P^{s,q}_{j,i}}{I^{s,q}_i}, \frac{P^{s,q}_{j,i}}{I^{s,q}_{j,i}})\big); q = 1, \ldots, c^s_i\}$, and $\boldsymbol{\mathcal{R}}^n_{(j,i)} = \{R^{n,q}_{j,i} = f\big(\min(\frac{P^{n,q}_{j,i}}{I^{n,q}_i}, \frac{P^{n,q}_{j,i}}{I^{n,q}_{j,i}})\big); q = 1, \ldots, c^n_i\}$, $\forall j = 1, \ldots, K_i, \forall i = 1, \ldots, N$. Each AP_i can use only one channel $Ch_i \in \{\boldsymbol{\mathcal{C}}^f_i \cup \boldsymbol{\mathcal{C}}^s_i \cup \boldsymbol{\mathcal{C}}^n_i\}$ to serve its users. Ch_i provides an actual rate $R^{Ch_i}_{j,i}$ between $AU_{j,i}$ and AP_i, where $R^{Ch_i}_{j,i} \in \{\boldsymbol{\mathcal{R}}^f_{(j,i)} \cup \boldsymbol{\mathcal{R}}^s_{(j,i)} \cup \boldsymbol{\mathcal{R}}^n_{(j,i)}\}$. According to this, we want to define the variable $X_{j,i}$ as

follows:

$$X_{j,i} = \begin{cases} 1 & \text{, if } R_{j,i}^{Ch_i} \geq R_{j,i} \\ 0 & \text{, otherwise} \end{cases},$$

where if $X_{j,i} = 1$ (or 0), this means that by choosing the channel Ch_i at AP_i, the $AU_{j,i}$ can (or cannot) be served by AP_i on Ch_i. It is important to define another type of variables as follows:

$$c_i^{r,q} = \begin{cases} 1 & \text{, if } Ch_i = C_i^{r,q} \\ 0 & \text{, otherwise} \end{cases}, \forall r \in \{f, s, n\}.$$

Table 5.1 summarizes these important notations.

5.3 Problem Formulation and Channel Assignment Solution

The objective function: This is a distributed system where each AP updates its channel selection only via passive sensing. The main objective at AP_i is to find the channel with the maximum number of served AUs while having the highest IN value. To protect neighbors from AP_i's interference, this channel should be chosen from the set $\boldsymbol{\mathcal{C}}_i^n$. If this is not possible, then it should be chosen from the set $\boldsymbol{\mathcal{C}}_i^s$, and if this is not possible, then it should be chosen from the set $\boldsymbol{\mathcal{C}}_i^f$; otherwise, all transmissions are blocked at AP_i. Note that the set $\boldsymbol{\mathcal{C}}_i^n$ has higher priority over both sets $\boldsymbol{\mathcal{C}}_i^s$ and $\boldsymbol{\mathcal{C}}_i^f$ for the chosen channel to be taken from. Also, the set $\boldsymbol{\mathcal{C}}_i^s$ has higher priority over the set $\boldsymbol{\mathcal{C}}_i^f$.

The problem formulation at AP_i can be derived as a three-stage optimization problem as given below:

Stage 1:

$$\max_{Ch^i \in \boldsymbol{\mathcal{C}}_i^n} \left\{ A_1 \sum_{j=1}^{K_i} X_{j,i} + \sum_{q=1}^{c_i^n} I_i^{n,q} \times c_i^{n,q} \right\} \tag{5.2}$$

s.t.

- $\sum_{q=1}^{c_i^n} c_i^{n,q} \leq 1$; (exclusivity/blocking constraint)

- $c_i^{n,q} = \begin{cases} 1 & \text{, if } Ch_i = C_i^{n,q} \\ 0 & \text{, otherwise} \end{cases}, \forall q = 1, \ldots, c_i^n$
- $X_{j,i} = \begin{cases} 1, & \text{if } R_{j,i} \leq \sum_{q=1}^{c_i^n} R_{j,i}^{n,q} \times c_i^{n,q} \\ 0, & \text{otherwise} \end{cases}, \forall j = 1, \ldots, K_i$
- $R_{j,i}^{n,q} = f\big(\min(\frac{P_{j,i}^{n,q}}{I_i^{n,q}}, \frac{P_{j,i}^{n,q}}{I_{j,i}^{n,q}})\big); q = 1, \ldots, c_i^n, \forall j = 1, \ldots, K_i;$

where $A_1 = c_i^n \times \max(\boldsymbol{\mathcal{I}}_i^n)$. Note that A_1 is treated as a very large number, where it plays a vital role in biasing the first stage optimization toward maximizing the number of AUs at AP_i, which is the main objective (since maximizing the number of served AUs and having the maximum IN value are contradicting objectives). At this stage, AP_i finds the channel from the set $\boldsymbol{\mathcal{C}}_i^n$ that maximizes the number of served AUs while having the maximum IN value at AP_i among all other channels from the same set. If the first stage is not successful in finding a channel that serves at least one AU from the AUs of AP_i, the optimization process goes to the second stage.

Symbol	Definition
$\boldsymbol{\mathcal{AP}}$	Set of all APs
$\boldsymbol{\mathcal{OSD}}$	Set of all OSDs
$\boldsymbol{\mathcal{AU}}_i$	Set of all AUs of AP_i
$\boldsymbol{\mathcal{C}}_i^f$	Set of first tier neighbors' channels of AP_i
$\boldsymbol{\mathcal{C}}_i^s$	Set of the second tier neighbors' channels of AP_i
$\boldsymbol{\mathcal{C}}_i^n$	Set of the channels not in $\boldsymbol{\mathcal{C}}_i^f$ or in $\boldsymbol{\mathcal{C}}_i^s$ of AP_i
$\boldsymbol{\mathcal{I}}_i^f$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^f$ at AP_i
$\boldsymbol{\mathcal{I}}_i^s$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^s$ at AP_i
$\boldsymbol{\mathcal{I}}_i^n$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^n$ at AP_i
$\boldsymbol{\mathcal{I}}_{(j,i)}^f$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^f$ at $AU_{j,i}$
$\boldsymbol{\mathcal{I}}_{(j,i)}^s$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^s$ at $AU_{j,i}$
$\boldsymbol{\mathcal{I}}_{(j,i)}^n$	Set of IN values on each channel in $\boldsymbol{\mathcal{C}}_i^n$ at $AU_{j,i}$
$\boldsymbol{\mathcal{R}}_i$	Set of needed rates by each AU$-AP_i$ connection
$\boldsymbol{\mathcal{P}}_{(j,i)}^f$	Received power set at AP_i on $\boldsymbol{\mathcal{C}}_i^f$ channels from $AU_{j,i}$
$\boldsymbol{\mathcal{P}}_{(j,i)}^s$	Received power set at AP_i on $\boldsymbol{\mathcal{C}}_i^s$ channels from $AU_{j,i}$
$\boldsymbol{\mathcal{P}}_{(j,i)}^n$	Received power set at AP_i on $\boldsymbol{\mathcal{C}}_i^n$ channels from $AU_{j,i}$
$\boldsymbol{\mathcal{R}}_{(j,i)}^f$	Actual rate set on channels $\boldsymbol{\mathcal{C}}_i^f$ between $AU_{j,i}$ and AP_i
$\boldsymbol{\mathcal{R}}_{(j,i)}^s$	Actual rate set on channels $\boldsymbol{\mathcal{C}}_i^s$ between $AU_{j,i}$ and AP_i
$\boldsymbol{\mathcal{R}}_{(j,i)}^n$	Actual rate set on channels $\boldsymbol{\mathcal{C}}_i^n$ between $AU_{j,i}$ and AP_i
Ch_i	The chosen channel for transmission at AP_i

Table 5.1. Notations summary

Stage 2: The optimization problem at Stage 2 is similar to the one in Stage 1 but the superscript changes from "n" to "s", since the need is to assign the channels from set $\boldsymbol{\mathcal{C}}_i^s$ instead of $\boldsymbol{\mathcal{C}}_i^n$. Note that A_1 should be replaced by $A_2 = c_i^s \times \max(\boldsymbol{\mathcal{I}}_i^s)$. If the second stage is not successful in finding a channel that serves at least one AU from the AUs of AP_i, the

optimization process goes to the third stage.

Stage 3: The optimization problem at Stage 3 is similar to the one in Stage 1 but the superscript changes from "n" to "f", since the need is to assign the channel from set $\boldsymbol{\mathcal{C}}_i^f$ instead of $\boldsymbol{\mathcal{C}}_i^n$. Note that A_1 should be replaced by $A_3 = c_i^f \times \max(\boldsymbol{\mathcal{I}}_i^f)$. Finally, if the third stage is not successful in finding a channel that serves at least one AU from the AUs of AP_i, the optimization process terminates and reports the blocking of all AUs (i.e., there is no chosen channel and $\sum_{q=1}^{c_i^n} c_i^{n,q} + \sum_{q=1}^{c_i^s} c_i^{s,q} + \sum_{q=1}^{c_i^f} c_i^{f,q} = 0$).

Since the total number of supported channels (c) by the system is small (e.g. 11 channels) and the number of AUs per AP is usually small, then this optimization problem can be solved optimally as in Algorithm 1, where steps 10 to 22 represent its core idea. These steps scan all the channels, in the set $\boldsymbol{\mathcal{C}}_i^n$, and pick the channel that can support the maximum possible number of AUs. If more than one channel is able to support the same maximum number, then the selection is for the one with the highest IN value (i.e., steps 16 to 21). An AU is assumed to be supported by a given channel if its $R_{j,i}$ is less than or equal to $R_{j,i}^{n,q}$ (i.e., step 13 and 14).

Remark. *Algorithm 1 is able to find the optimal channel assignment at AP_i in linear time with complexity $\mathcal{O}(K_i)$.*

At the worst scenario, AP_i checks all frequency channels supported by the system (i.e., c channels). Without loss of generality, it can be assumed that all the frequency channels are in the set $\boldsymbol{\mathcal{C}}_i^n$ (i.e., $c_i^n = c$). Hence, for each channel, the algorithm checks the transmission possibility for each AU of AP_i (i.e., all K_i AUs); consequently, the worst time complexity is $\mathcal{O}(cK_i)$. But the value of c in the WiFi system is a small constant. This makes the time complexity of Algorithm 1 $\mathcal{O}(K_i)$. With this time complexity, Algorithm 1 is able to find the optimal channel assignment in linear time, where the optimal channel assignment is for the channel that can serve the maximum number of the AUs from $\boldsymbol{\mathcal{AU}}_i$ and has the highest IN value among the channels that can support the same number of AUs.

Algorithm 2 AP_i GTCA channel assignment pseudo code

1: **Inputs:**
2: $\boldsymbol{\mathcal{AU}}_i$, K_i, $\boldsymbol{\mathcal{R}}_i$, $\boldsymbol{\mathcal{C}}_i^f$, $\boldsymbol{\mathcal{C}}_i^s$, $\boldsymbol{\mathcal{C}}_i^n$, $\boldsymbol{\mathcal{I}}_i^f$, $\boldsymbol{\mathcal{I}}_i^s$, $\boldsymbol{\mathcal{I}}_i^n$, $\boldsymbol{\mathcal{I}}_{(j,i)}^f$, $\boldsymbol{\mathcal{I}}_{(j,i)}^s$, $\boldsymbol{\mathcal{I}}_{(j,i)}^n \forall j$
3: **Generate:**
4: $\boldsymbol{\mathcal{P}}_{(j,i)}^f$, $\boldsymbol{\mathcal{P}}_{(j,i)}^s$, $\boldsymbol{\mathcal{P}}_{(j,i)}^n$, $\boldsymbol{\mathcal{R}}_{(j,i)}^f$, $\boldsymbol{\mathcal{R}}_{(j,i)}^s$, $\boldsymbol{\mathcal{R}}_{(j,i)}^n$ $\forall j$
5: **Initialization:**
6: Let the number of served AUs $(NSAU) \leftarrow 0$ and assume a variable $NSAU_{Old} \leftarrow 0$
7: Let $Ch_i \leftarrow Void$
8: Assume a variables $I \leftarrow -\infty$ and $I_{Old} \leftarrow -\infty$
9: **procedure** CHANNEL ASSIGNMENT
10: **for** $(q \leftarrow 1$ to $c_i^n)$ **do**
11: $I \leftarrow I_i^{n,q}$
12: **for** $(j \leftarrow 1$ to $K_i)$ **do**
13: **if** $(R_{j,i} \leq R_{j,i}^{n,q})$ **then** $NSAU \leftarrow NSAU + 1$
14: **end if**
15: **end for**
16: **if** $(NSAU > NSAU_{Old})$ or $((NSAU = NSAU_{Old})$ and $(I > I_{Old}))$ **then**
17: $I_{Old} \leftarrow I$
18: $NSAU_{Old} \leftarrow NSAU$
19: **if** $(NSAU > 0)$ **then** $Ch_i \leftarrow C_i^{n,q}$
20: **end if**
21: **end if**
22: **end for**
23: **if** $(NSAU = 0)$ **then**
24: Replace the superscript "n" by "s" and repeat steps 10-22
25: **end if**
26: **if** $(NSAU = 0)$ **then**
27: Replace the superscript "n" by "f" and repeat steps 10-22
28: **end if**
29: **if** $(NSAU = 0)$ **then**
30: All transmissions at AP_i are blocked and $Ch_i \leftarrow Void$
31: **end if**
32: **end procedure**

5.4 Channel Access Protocol

This section presents the GT-MAC protocol, which is a new MAC protocol based on the infra-structure based IEEE 802.11 MAC protocol. This MAC protocol uses the GTCA channel assignment mechanism described previously, instead of the random channel assignment used in the current IEEE 802.11 protocol. Since the GT-MAC is based on the IEEE 802.11 protocol, it inherits the following features:

1. There is no dedicated communication between the APs.
2. Each AP keeps broadcasting beacons on its currently specified channel using a fixed power level. These beacons contain information about the AP such as the network identification (ID) (i.e., SSID).
3. Each user scans the WiFi frequency spectrum and picks the authorizing AP that has the best signal strength.
4. The use of a staircase Data Rate-SINR relationship.

To overcome the limitation of the Random channel assignment in the current IEEE 802.11, the proposed GT-MAC protocol must have the following features to achieve this goal.

1. The channel assignment is fully automated.
2. The chosen channel should not disturb the transmissions of the neighboring APs as much as possible.
3. The chosen channel must satisfy the needed AU-AP link data rate.
4. The channel assignment must not choose the best channel, but just the channel that satisfies the needed rate by the AU-AP transmissions.

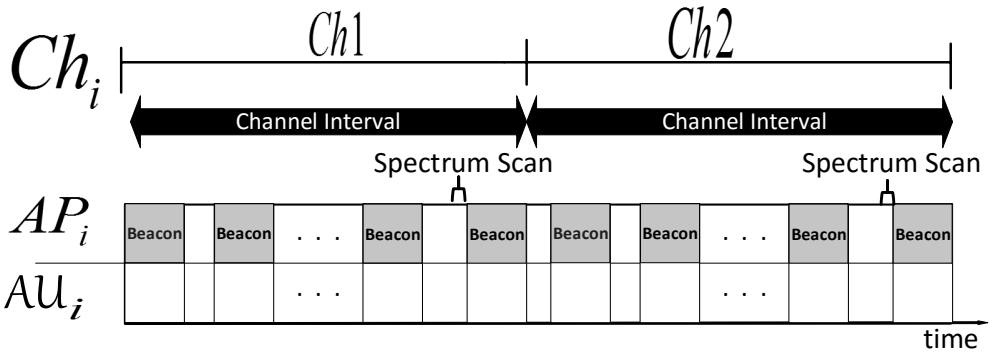


Figure 5.2. Beaconing Mode.

5.4.1 GT-MAC Operation

In the GT-MAC protocol, there are two modes of operation for the AP: The Beaconing mode and the Date transmission mode.

1- Beaconing Mode: Fig. 5.2 describes the packet transmission operation of this mode.

Fig. 5.2 in this mode shows that AP_i keeps sending beacon packets (beacons) to its users to keep the connectivity and to inform its first tier neighbors about its first tier AP neighbors' ID list (FHID) and their channels of operation (i.e., $\boldsymbol{\mathcal{C}}_i^f$). Note that this kind of information sharing is not considered direct communication between the APs, since there is no back and forth packet transfer between specified (transmitter and receiver) APs. The transmitter AP is just broadcasting its information, and the receiver APs are using the passive learning to acquire this information. Also note that the GT-MAC requires the addition of the FHID and $\boldsymbol{\mathcal{C}}_i^f$ lists to the beacon of AP_i, unlike the 802.11 that lacks this kind of information. The added overhead, because of these lists, can be considered negligible, since such information has small bit-size. The time between two consecutive beacons is known as the "beacon interval" (BI). The value of BI is configurable and it is usually 100 ms in practice, [54]. From Fig. 5.2, the "channel interval" (CI) is a new configurable parameter that represents the number of beacons that must be transmitted on the current channel before switching to the new channel using GTCA. Just before the last beacon in the current channel

interval, the AP makes a spectrum scan and runs the GTCA channel assignment mechanism to find its next beaconing channel. The spectrum scan is also important in updating the FHID and second tier neighbor ID list (SHID) of AP_i. After that, the AP broadcasts its next beaconing channel along with its FHID and $\boldsymbol{\mathcal{C}}_i^f$ on the last beacon of the current beaconing channel, then starts a new CI on the next beaconing channel. Furthermore, all the AUs, of this AP, also move to this new beaconing channel.

Remarks:

- It is possible to have the next beaconing channel be the same as the current beaconing channel.

- Each beacon has a 1-bit field called BM. If $BM = 1$, then this informs the AUs of the AP that the system is on the beaconing mode. But if $BM = 0$, then this indicates that the operation switched to data transmission mode.

2- Data Transmission Mode: This mode is described by a simple example shown in Fig. 5.3. This figure shows a random time snapshot from the network operation. This snapshot starts with a beacon from AP_i at time t_1 with $BM = 1$, to indicate that the AP cell is on the beaconing mode. Then at a random time t_2, between t_1 and $t_1 + BI$ (i.e. the next scheduled beaconing time), $AU_{2,i}$ gets some data to transmit to AP_i. As such, $AU_{2,i}$ senses the power on the currently used beaconing channel (i.e., $Ch^{'}$) and finds it clear. Hence, $AU_{2,i}$ starts a random back-off counter to compete on the channel. Then the counter ends, which allows $AU_{2,i}$ to send an RTS packet to AP_i. On its turn, AP_i responds to $AU_{2,i}$ with a CTS packet. This handshaking process (i.e., the RTS-CTS control packets exchange) starts a time AW that contains T TSs ($\{TS_1, \ldots, TS_T\}$). Each TS has two time periods for the RTS-CTS exchange. Note that the first TS (i.e., TS_1) is used by the transmitter that first initiated the AW (here it is $AU_{2,i}$).

At t_4 (the starting time of TS_2), AP_i has some data to transmit to $AU_{1,i}$. Hence, it competes like any other device in the network on TS_2 by starting a random back-off counter.

Since its timer finishes first, it gets the right to use TS_2. The figure shows that AP_i wins TS_2, and it sends an RTS to $AU_{1,i}$, which sends back a CTS at the second period of TS_2. This competition on the frequency channel $Ch^{'}$ keeps going between the AUs and the AP to the rest of the TSs in the AW, where the last TS ends at time t_9.

The RTS and CTS packets from the AP contains the AP ID (i.e., AP_i), the AU ID (i.e., $AU_{j,i}$), FHID and their channels (i.e., $\boldsymbol{\mathcal{C}}_i^f$), and the needed rate for transmission $R_{j,i}$. The AU RTS and CTS packets contain $AU_{j,i}$, AP_i, $R_{j,i}$, and list of the IN value on each channel in the system (i.e., the set $\boldsymbol{\mathcal{I}}_{(j,i)}^f \cup \boldsymbol{\mathcal{I}}_{(j,i)}^s \cup \boldsymbol{\mathcal{I}}_{(j,i)}^n$). This extra information added to the RTS and CTS also has a negligible effect on the overhead because of its small bit-size.

After getting all the pieces of information about the needed transmissions, AP_i runs the GTCA channel assignment (Algorithm 1) and chooses the channel that satisfies the optimization problem. Note that the handshaking process is also important, because it allows the AP to sense the received power from the AU and to estimate the power received at AU. Hence, building the sets $\boldsymbol{\mathcal{P}}_{(j,i)}^f$, $\boldsymbol{\mathcal{P}}_{(j,i)}^s$, and $\boldsymbol{\mathcal{P}}_{(j,i)}^n$ as described previously.

After deciding Ch_i, AP_i sends in a special beacon the transmissions that are allowed to proceed and the transmissions that are blocked, according to the result from Algorithm 1. This beacon has the bit $BM = 0$ to indicate that the beaconing mode ended and the data transmission mode started. This beacon contains AP_i ID, the ID list of the AUs, which transmissions are allowed to proceed, FHID of AP_i, Ch_i ($Ch_i = Ch^{''}$ in our example), and $\boldsymbol{\mathcal{C}}_i^f$. Note that all of these operations happen on the old channel (i.e., on $Ch^{'}$). After this, AP_i and its AUs move their frequency of operation to the new channel (i.e., $Ch^{''}$) and start the data transmission mode.

After the end of data transmission mode, the AP senses all the channels again and updates its FHID list, $\boldsymbol{\mathcal{C}}_i^f$, its SHID, $\boldsymbol{\mathcal{C}}_i^s$, $\boldsymbol{\mathcal{C}}_i^n$, $\boldsymbol{\mathcal{I}}_i^f$, $\boldsymbol{\mathcal{I}}_i^s$, and $\boldsymbol{\mathcal{I}}_i^n$. Then AP_i makes a channel assignment using Algorithm 1 to find the next value of Ch_i that satisfies its next operation (which is the beaconing mode), then it sends a beacon (that contains the new value of Ch_i) on the current channel (i.e., $Ch^{''}$) with $BM = 1$ to indicate the start of the beaconing mode

and the end of the transmission mode. After this, the AP and all of its AUs move their operation frequency to the new channel specified in this beacon.

Note that in the data transmission mode the AP and the AUs use the CSMA/CA before sending their data packets. This is important mainly to prevent the collision with the surrounding cells that may use the same channel. Using the CSMA/CA in the data mode makes the GT-MAC control packet overhead (at the worst case) double the overhead of the IEEE 802.11. This is because the RTS-CTS exchange is needed on the AW and the data transmission mode. The worst case (in terms of overhead) happens when all the requested transmissions are approved by the AP.

On the data transmission mode, the BM value is 0, which means neither the AP nor any of its AUs can start an AW and only the approved transmissions can use the chosen channel until BM changes to 1.

It is assumed in this MAC protocol that the beaconing operation requires the least data rate provided by the system, since it is all about transmitting the control packets (i.e., beacon, RTS and CTS packets). This assumption is very important in identifying the next beaconing channel.

5.4.2 Knowing the second tier neighbors (SHID list)

The APs know their second tier neighbors by listening to the announcements from other APs. For example, if AP_1 has AP_2 as its first tier neighbor, then when AP_2 announces AP_3 as one of its first tier neighbors, AP_1 knows that AP_3 is a second tier neighbor. This idea has been used previously in the literature, as in [8], and [31]. The consideration of second tier neighbors is important to protect the AUs at the cell edge from experiencing interference from their AP's second tier neighbors.

5.4.3 GT-MAC scalability and deciding the beaconing channel at the system setup

If a new AP (e.g. AP_i) is deployed in the vicinity of already existing APs, then AP_i starts its operation by scanning the spectrum. Since AP_i is at its first instant of operation,

it has no information about its AUs (i.e., the $\boldsymbol{\mathcal{I}}^f_{(j,i)} \cup \boldsymbol{\mathcal{I}}^s_{(j,i)} \cup \boldsymbol{\mathcal{I}}^n_{(j,i)}, \forall j = 1, \ldots, K_i$ and how far away they are), then Algorithm 1 can not be used because of the lack of information. As such, AP_i senses the spectrum to build the $\boldsymbol{\mathcal{C}}^f_i$, $\boldsymbol{\mathcal{C}}^s_i$, $\boldsymbol{\mathcal{C}}^n_i$, $\boldsymbol{\mathcal{I}}^f_i$, $\boldsymbol{\mathcal{I}}^s_i$ and $\boldsymbol{\mathcal{I}}^n_i$. After that, AP_i sends its first beacon on the best channel (i.e., the one with the least IN value) from the set $\boldsymbol{\mathcal{C}}^n_i$. This beacon is called the recognition beacon (RB), and it is used to let the AUs and other APs know about the setup of AP_i. The RB contains the ID of the AP and a request to the AUs to send their information (i.e., $\boldsymbol{\mathcal{I}}^f_{(j,i)} \cup \boldsymbol{\mathcal{I}}^s_{(j,i)} \cup \boldsymbol{\mathcal{I}}^n_{(j,i)}, \forall j = 1, \ldots, K_i$) to the AP. The RB also contains the number of TSs (i.e., T) per AW. After sending the RB, an AW opens with T TSs. Then the AUs compete on each TS in order to send their information to the AP. It is worth mentioning that each of these TSs contains only one time period. This time period is used to send the RTS packet from the AU to the AP with no need for a CTS back, since the AP is only collecting information.

Remarks: The values of the BI, CI, and T depend on the network dynamics, and it is more efficient to automate their values. However, in this work they are left to be user defined parameters. This is because changing them dynamically may jeopardize the speed of the protocol convergence. As part of our future work, we will work on how to automate the change of the BI, CI, and T while having a marginally negative effect on the protocol speed. Our future work may also contain a collaborative channel assignment between the APs, related to the same administrative entity, and how they interact with other AP clusters for a better throughput enhancement. Another interesting aspect to be considered is to provide a mechanism to limit the effect of compromised AUs that ask for data rates higher than what they really need.

5.5 Performance Evaluation

5.5.1 Simulation Setup

We consider N APs deployed randomly in a 200m $\times$ 200m area. These APs coexist with M OSDs. Each AP_i has K_i AUs with randomly generated positions in its coverage

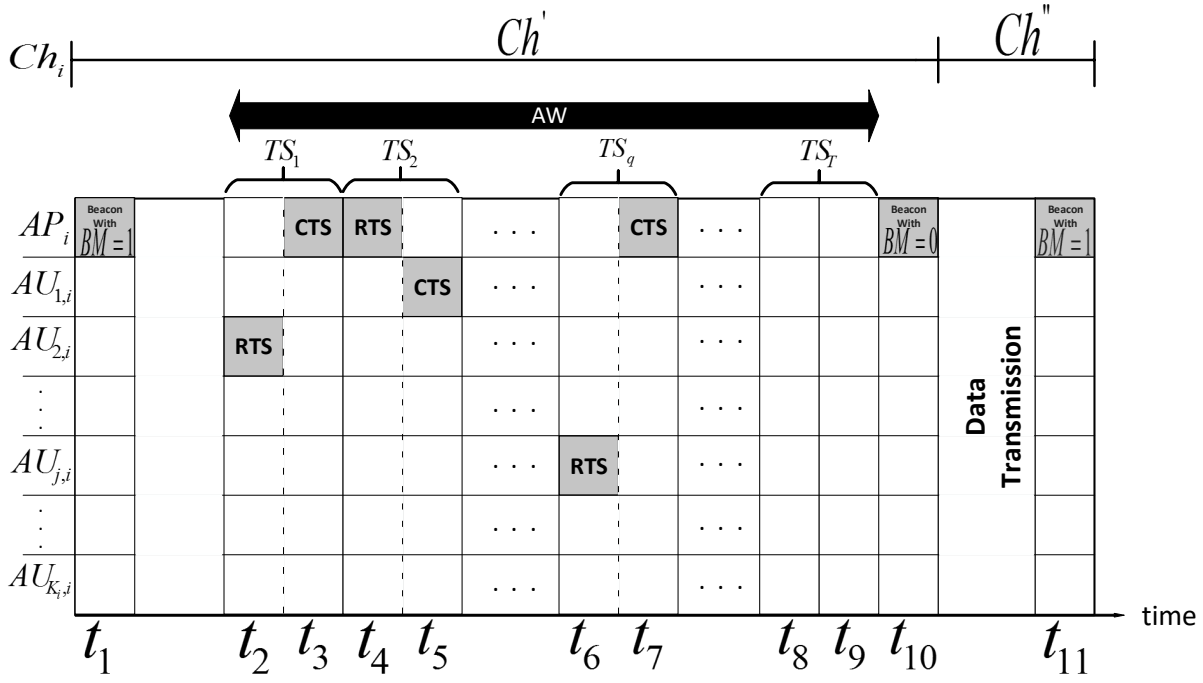
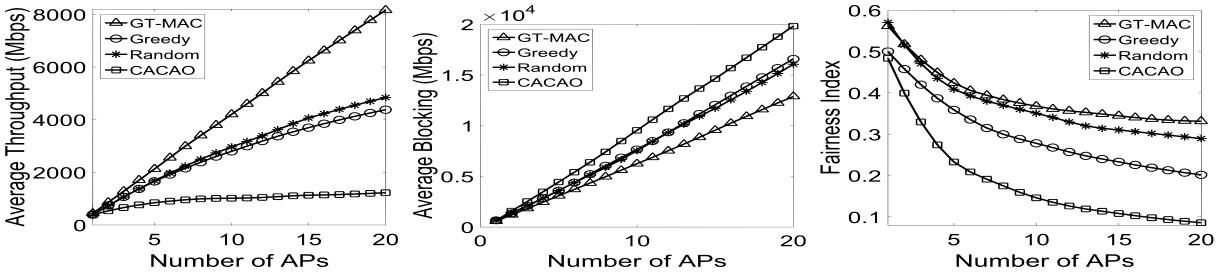


Figure 5.3. Time snapshot of the data transmission mode.

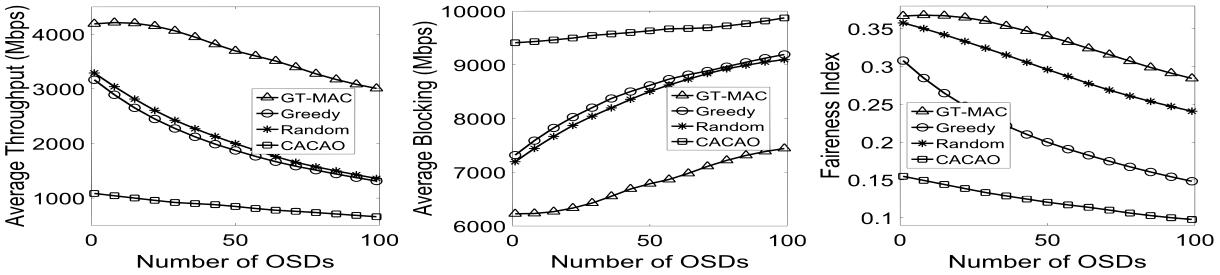
area. The coverage area for all APs is determined by the maximum radius r_{Max}, which is the maximum distance from the AP where the AUs are still be able to receive from it (i.e., the AP). r_{Max} can be calculated using Equation (5.1), the value of the shortest wavelength used in the system, and μ^*, which is the SINR threshold of the device. The performance comparison is made between the GT-MAC, the Greedy [68], the Random [4], and the CACAO [80] channel assignment based MAC protocols[1]. The CACAO is a simple, distributed and scalable channel assignment with a novel idea of predicting the future interference levels. By using the CACAO, the AP chooses the channel with the least predicted future interference. The interference level is predicted on a given channel by summing all the current sensed data-rates (by the AP's cell) from other cells on this channel with the future needed data rates of the AP's cell. The main limitation of this channel assignment, however, is the ignoring of the current interference between the AP cells and the interference coming from

1 As aforementioned, other recent works often use very specific models, such as assuming a centralized controller or active collaborations among APs. Therefore, for fair comparison, we choose the CACAO channel assignment based MAC protocol, which is one of the most recent works that use a general comprehensive system model similar to ours.



(a) AANT Vs N. (b) ABNT Vs N. (c) FI Vs N.

Figure 5.4. $M = 10$,$K_{Max} = 5$,$c = 11$.



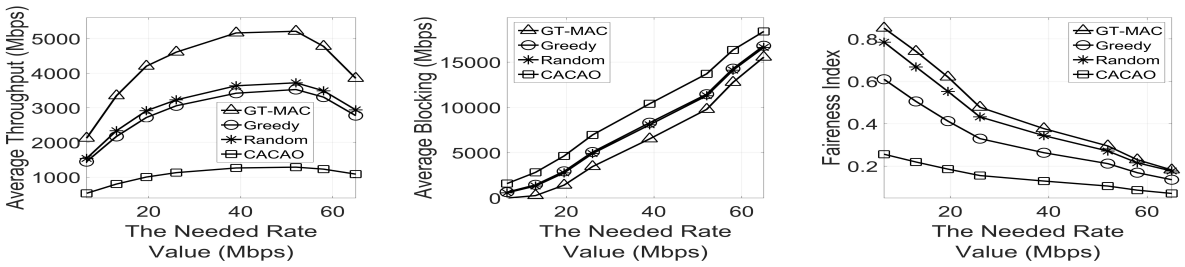
(a) AANT Vs M. (b) ABNT Vs M. (c) FI Vs M.

Figure 5.5. $N = 10$, $K_{Max} = 5$, $c = 11$

the OSDs. This limitation may cause the CACAO not to perform well when there are OSDs interfering with the system, as shown in later results. Considering the current interference from other AP cells and OSDs is crucial, since there is no connection between the APs, each AP and its AUs know the data rates of other AP cells from passive learning. Then if the current interference from other AP cells and OSDs is very high, the AP and its AUs are not able to read the received signals (from these cells) and know their data rates. Therefore, the predicted future interference, on a channel with interference higher than the threshold may be considered zero while in reality it has a high level of interference. As a result, the AP may choose this channel for transmission (since it has zero predicted interference according to the CACAO), which makes the problem even worse.

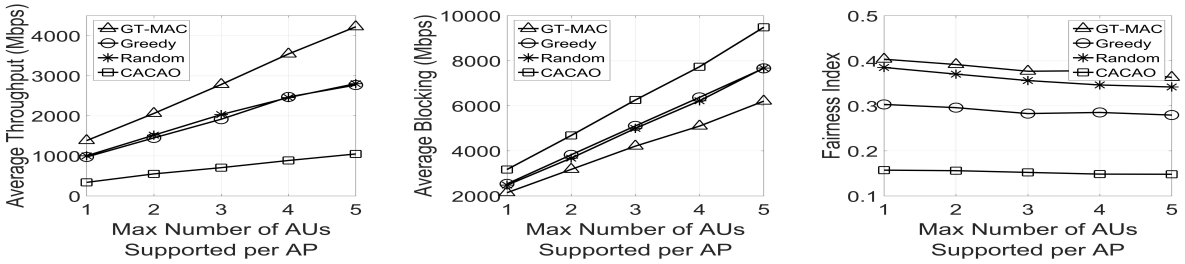
The performance analysis is conducted using the $MATLAB^{®}$ simulations while using the overall network throughput and Jain's fairness index (FI) [43] as the main metrics.

The presented results are based on the use of the staircase SINR-Data rate relation of the IEEE 802.11n (given in Table 5.2), the $2.4GHz$ band (using the $5GHz$ is optional in the IEEE 802.11n), -90dBm average noise power, $G_t = G_r = 1$dB, $\mu^* = 20$dB, $P_t = 12$dBm,



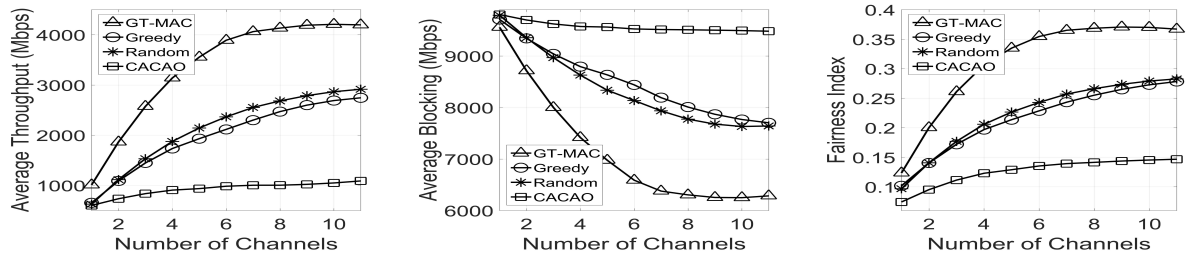
(a) AANT Vs $R^{'}$. (b) ABNT Vs $R^{'}$. (c) FI Vs $R^{'}$.

Figure 5.6. $N = 10$, $M = 10$, $K_{Max} = 5$, $c = 11$



(a) AANT Vs K_{Max}. (b) ABNT Vs K_{Max}. (c) FI Vs K_{Max}.

Figure 5.7. $N = 10$,$M = 10$,$c = 11$



(a) AANT Vs c. (b) ABNT Vs c. (c) FI Vs c.

Figure 5.8. $N = 10, M = 10, K_{Max} = 5$

$\alpha = 3.5$ and $d_o = 1$m. The elements in the set $\boldsymbol{\mathcal{R}}_i$ are generated randomly from the set $\{6.5, 13, 19.5, 26, 39, 52, 58.5, 65\}$ except for the results in Fig. 5.6 where the AUs are forced to require specific rate.

SINR (dB)	< 20	20-22	22-24	24-26	26-28	28-30	30-32	32-34	$34 <$
Data Rate (Mbps)	0	6.5	13	19.5	26	39	52	58.5	65

Table 5.2. IEEE 802.11n SINR-Data rate formula.

5.5.2 Simulation Results

Figs. 5.4(a) and 5.4(b) show the average achieved network throughput (AANT) and average blocked network throughput (ABNT), respectively, as functions of the APs number (N). These figures show that the GT-MAC outperforms all other channel assignments. The GT-MAC provides an increase in the AANT by up to 67.5%, 88.5%, and 559% compared to the Random, the Greedy, and the CACAO based MAC protocols, respectively. These figures also show that as the number of APs increases, more spectrum requests are generated. Some of these requests may have the chance to succeed which increases the AANT, while other requests are blocked, which increases the ABNT.

Figs. 5.5(a) and 5.5(b) plot the AANT and ABNT as functions of the OSDs number (M), respectively. As it is obvious, as M increases, the number of interference-free channels decreases. Hence, the AANT decreases, and the ABNT increases. Fig. 5.5(a) also shows that GT-MAC outperforms the Random, the Greedy, and the CACAO by up to 86.5%, 102%, and 355%, respectively.

In Figs. 5.6(a) and 5.6(b), we are interested in examining the AANT and ABNT as functions of the rate demand ($R^{'}$), where $R_{j,i} = R^{'}, \forall j = 1, \ldots, K_i$ and $\forall i = 1, \ldots, N$. These figures show that both AANT and ABNT increase as functions of $R^{'}$. This is because the higher the demanded rate, the higher the AANT value, if this rate is approved by the AP. The AANT keeps increasing with $R^{'}$ up to an optimal point, then starts to decrease, since the system cannot handle the increase in the demanded rate. If the rate is blocked, then

this increases the ABNT, and, hence, the higher the $R^{'}$ the higher the ABNT. Fig. 5.6(a) shows that the GT-MAC outperforms the Random, the Greedy, and the CACAO based MAC protocols by up to 38%, 50%, and 311%, respectively.

Figs. 5.7(a) and 5.7(b) plot the AANT and ABNT as functions of the maximum number of AUs supported by any AP (K_{Max}). These figures show the increase in both the AANT and ABNT as K_{Max} increases. This is because as K_{Max} increases, more rate requests are generated, which increases the AANT if these requests are approved and increases the ABNT if these requests are blocked. Fig. 5.7(a) shows that the GT-MAC outperforms the Random, the Greedy, and the CACAO based protocols by up to 51%, 52.5%, and 303%, respectively. To study the performance of GT-MAC under the change of the channels number (c), Figs. 5.8(a) and 5.8(b) plot the AANT and ABNT as functions of c. It is clear that the increase in c provides more channel opportunities. Hence, more rate requests are processed and less are blocked, which results in increasing AANT and decreasing ABNT. Fig. 5.8(a) shows that GT-MAC outperforms the Random, the Greedy, and the CACAO MAC protocols by up to 68%, 80%, and 323%, respectively. Figs. 5.4(c), 5.5(c), 5.6(c), 5.7(c), and 5.8(c) plot the fairness index as a function of N, M, $R^{'}$, K_{Max}, and c, respectively. These figures show that the GT-MAC outperforms all the other channel assignments. It is worthwhile to mention that the fairness index is calculated by $FI = \frac{(\sum_{i=1}^{N} x_i)^2}{N\sum_{i=1}^{N} x_i^2}$, where x_i is the total throughput of AP_i cell. In all the AANT and ABNT figures, we notice that the Greedy MAC protocol is slightly outperformed by the Random MAC protocol. This is because, in the Greedy, the first deployed APs in the system always take the best channels for their transmissions even though they may require low data rates. On the other hand, this is not the case with the Random, since being the first deployed AP does not mean getting the best channel, which gives better opportunity for other APs in getting good channels for their transmissions. We also notice that the GT-MAC performs the best. This is because the GT-MAC not only avoids the situation of being greedy, but also chooses the channel generously and wisely, so that more traffic demands can be satisfied, resulting in much higher overall

throughput and better fairness.

It is worth mentioning that we have conducted similar simulations using the properties of the IEEE 802.11b (see [47]) and IEEE 802.11g standards, and similar conclusions were made.

5.6 Conclusion

In this chapter, we presented a new throughput oriented IEEE 802.11 based MAC protocol for infra-structured WiFi networks. The novelty of this protocol comes from its channel assignment, where the channel with the least SINR but satisfying the needed data rate, is chosen. The other part of the novelty comes from the high level of protection to the first and second tier neighbors. The GT-MAC has several advantages, including simplicity, similar control packet overhead to the IEEE 802.11, accounting for the effect of the OSDs, dynamic and fully automated channel assignment, no restriction on the density or positions of the APs, no need for active collaboration, the sufficiency of passive learning for the channel assignment, and the high level of compatibility with the IEEE 802.11 standards. The experiment results show that the GT-MAC outperforms the Random, the Greedy, and the CACAO channel assignment based MAC protocols by up to 86.5%, 102%, and 559% in terms of throughput, respectively. In terms of fairness, the GT-MAC achieved the best fairness performance.

CHAPTER 6

Coexistence Channel Assignment Mechanisms For WiFi and LTE Over The Unlicensed Band

In the first four chapters of this dissertation, we have presented multiple system models to improve the spectrum sensing abilities of wireless networks. Having a good spectrum sensing ability is important for heterogeneous networks to coexist on the same frequency band. One important example on this coexistence is the LTE-WiFi coexistence on the unlicensed band. Spectrum sensing will enable the LTE cell to detect any WiFi transmissions within the cell and vice versa. For the LTE cell to detect a WiFi device (i.e., WiFi transmission), the LTE users [i.e., User Equipment devices(UEs)] act as sensors, the LTE eNB acts as a fusion center (FC), and the WiFi device acts as a Primary radio User (PU). The same analogy can be applied on the WiFi cell to detect an LTE device (i.e., LTE transmission). Having a robust and accurate spectrum sensing abilities is important at the Medium Access Control (MAC) layer. This will enable the MAC layer to generate accurate channel access decisions in order to prevent interference, collision, and get the best out of the available transmission opportunities to satisfy the network objective. In this chapter, we assume the LTE and WiFi devices are equipped with perfect spectrum sensing abilities and we concentrate on how to improve the MAC layer decision abilities in both the LTE and WiFi networks. This chapter studies the interactions between three types of networks on the unlicensed band in the presence of interference. These networks are: LTE network equipped with dynamic channel assignment mechanism, WiFi network equipped with dynamic channel assignment mechanism, and legacy WiFi networks that does not use dynamic channel assignment mechanisms but rather use either the random channel assignment or the greedy channel assignment [47], which are considered to be passive as to be explained later on.

In developing the proposed channel assignment mechanisms, we depend on the results and findings form Chapter 5 and [47], where the Generous Throughput-oriented Channel Assignment (GTCA) mechanism proved its ability in improving the coexistence, the throughput, and fairness within the WiFi network itself, hence, applying these findings on the LTE-WiFi coexistence problem promises good results.

Based on GTCA, we propose an enhanced version of it called the Enhanced-GTCA (EGTCA) which is much simpler and less conservative. We also propose a new version of the ordinary Greedy channel assignment mechanism (called the Enhanced-Greedy channel assignment mechanism) where the cell chooses the best channel that is currently available (i.e., while taking the current network activities in consideration).

The results show the effectiveness of the proposed channel assignment mechanisms in improving the throughput and fairness (i.e., coexistence) for each individual network and for the all the networks together.

6.1 Introduction

This chapter proposes new channel assignments based on the idea of GTCA (in [47]) and an enhanced version of the Greedy channel assignment mechanism. The main purpose is to address the problem of coexistence between the WiFi and the LTE networks on the unlicensed band. Recently, the research and industry community shed the light on this problem to address the increasing demand on spectrum and to put the new foundations for the fifth generation (5G) of broadband cellular networks. The LTE was proposed to work on the licensed band only. But recently, after adding the carrier aggregation capability to the LTE standard (i.e., the transmission can happen on multiple contiguous or non-contiguous channels at the same time), there has been an increasing interest in the unlicensed band to gain more throughput and coverage. Even though the advantages may be great for the LTE networks, some challenges must be addressed first. Some of these challenges are: 1) the unlicensed band should be shared with other technologies such as WiFi, 2) there should be

some harmony in accessing and abandoning the spectrum between these different technologies to reserve fairness among them, 3) meeting all the regulatory rules such as listen-before-talk (LBT), power control, maximum transmit power, etc. The LBT technique means that the channel cannot be accessed until it is free. The devices, that use LBT, sense the channel and access it only when it is free.

This work proposes new channel assignments for the LTE and the future WiFi protocols, these channel assignments are supposed to be able to address most of these challenges and achieve the objective of coexistence, fairness, and maximizing the throughput for both the LTE and WiFi systems.

There are few LTE system models proposed in the literature to address the coexistence issue. The most important proposals are LTE-U (LTE-unlicensed), LTE-eLAA (LTE-enhanced Licensed Assisted Access), MulteFire, and LWA (LTE WLAN Aggregation). The LTE-U is based on the LTE standard before Release 13. It uses carrier aggregation of licensed LTE and unlicensed bands. This proposal does not use the LBT technique which makes it not eligible for deployment in some parts of the world (e.g. Europe and Japan). In the LTE-U, the device scans the unlicensed spectrum for a free channel, if none is found, then it uses an on-off duty cycle on one of the currently occupied channels. The LTE-eLAA comes as an enhanced version (in 3GPP Release 14) of LTE-LAA (in 3GPP Release 13), the main difference is that LTE-LAA does not support two-way communications (i.e. only the down-link) on the unlicensed channel while LTE-eLAA does. Both have an anchor licensed frequency to communicate the control and data packets between the eNB and UEs while the unlicensed frequency is left for data packets only. The importance of the anchor frequency is to assure the reliability and the Quality of Service (QoS) of the system. Both the LAA and eLAA use LBT to access the unlicensed spectrum, which makes it eligible for deployment world wide. The LAA and eLAA devices use the LTE standard for transmission on the unlicensed channel once it gains the access to it. The MulteFire is a stand-alone LTE deployment over the unlicensed spectrum with no anchor licensed frequency. It has the

potential to compete with WiFi since it can be deployed anywhere and by anyone exactly like WiFi networks but with the use of LTE communication over the unlicensed spectrum instead of the WiFi communication, which is expected to provide higher data rates than WiFi. Finally, the LWA is very similar to eLAA but instead of sending the data over the unlicensed frequency using the LTE standard, the LWA sends it using the WiFi standard. This means that the packet is generated in the LTE format then it gets encapsulated in a WiFi MAC packet format and sent through the WiFi physical layer (i.e. according to the WiFi standard). The problem with the LWA compared to eLAA is that eLAA can achieve much higher throughput than LWA because eLAA uses LTE standard on the unlicensed spectrum which allows the eNB to serve multiple UEs simultaneously. On the other hand, LWA uses the WiFi protocol on the unlicensed spectrum to deliver WiFi-encapsulated LTE packets, this means the eNB can handle one UE at a time, hence, lower throughput is expected. LWA is expected to have similar throughput to the WiFi on the unlicensed spectrum but if we add the LTE throughput on the licensed spectrum, then it is expected for LWA to achieve higher throughput values compared to WiFi but still less than the case of using eLAA.

6.1.1 Related Work

In [77], the authors propose a dynamic channel access mechanism for LTE cells that coexist with WiFi cells on the unlicensed band (the LTE-eLAA form is adopted). This channel assignment takes into account the traffic load of each eLAA cell. The eLAA cell with low traffic load searches for an idle channel to transmit for a specific time. If the transmission not finished by the end of this time, the cell should move to another idle channel. On the other hand, if this cell has high traffic load, then it will occupy the channel most of the time. Hence, in this case (i.e. high traffic load), the eLAA cell needs to give the WiFi cells the opportunity to use the channel by leaving a specified number of resources blocks (i.e. time-frequency resources) unoccupied for the WiFi use.

In [59], the authors shed the light on the problem of cell identification between WiFi

and LTE on the unlicensed band. Since WiFi and LTE are totally different standards they cannot interpret each others' packets and extract the needed information about the selected channel and cell traffic load. This means they need to depend solely on energy detection. Nevertheless, the energy detection does not work well in all cases because of the difference in signal modulation or the received energy below the threshold that may be different for different technologies. This energy detection inefficiency may cause a lot of interference between the different systems. Hence, the authors propose two solutions for this problem. The first solution makes use of the LTE and WiFi modules that are available in any smart phones. Hence, the LTE and WiFi may depend on the other module to extract/broadcast the important information from/to the other system in the area of coverage. This solution is also compatible with legacy WiFi networks. But this solution suffers from two problems, first: it drains the batteries of the smart phones because of the regular spectrum scans done by the two technologies, second: multiple smart phones may be needed in order to detect all the neighboring cells. Hence, the authors proposed another solution that may overcome these problems. In this second solution, there are "friendly" WiFi APs connected to the LTE eNB. These AP are deployed at certain distances from the eNB to cover its whole coverage area. The mission of these APs is to broadcast/receive the important LTE/WiFi information to/from the surrounding WiFi cells. The WiFi APs also have "friendly" eNB that have similar functionality. Due to the different energy detection thresholds between the LTE and WiFi, a lot of collision between the two systems are expected. Hence, the authors propose an adaptive energy detection threshold mechanism in order to reduce the amount of collisions between the two systems.

In [34], the authors study the effect of imperfect sensing on the LTE-LAA network throughput while coexisting with the WiFi networks. They aim to maximize the throughput by optimizing the energy detection sensing duration and threshold. In this analytical study, the authors take into account the effect of energy detection sensing error, the change in the channel conditions during the sensing window, and the exponential backoff counter(called

CAT4 by 3GPP, see [59]) used by the WiFi and LAA system [59].

In [69], the authors propose a new resource allocation scheme at the eNB that assigns the resource blocks and their corresponding power values to the UEs in the cell with the objective to maximize the cell sum rate. This scheme takes into account the licensed and unlicensed bands at the same time, the imperfect spectrum sensing, and the MIMO spatial degrees of freedom. Exploiting the MIMO spatial degrees of freedom allows the eNB to transmit at the same channel of the neighboring WiFi simultaneously without any interference which does not contradict with the LBT requirements. The authors formulated the problem as a MINLP problem and then used the Lagrange Dual Decomposition Method to solve the problem optimally.

In [60], the authors model and analyze a heterogeneous network that consists of two different radio access technologies (RATs) (RAT-L and RAT-U). The network consists of K-tiers of APs (the tier is a set of APs of the same type), the first K-1 tiers use RAT-L while tier K uses RAT-U. The AP in the first K-1 tiers can access RAT-L channel without contention, while the APs in tier K have to contend in order to access the channel of RAT-U. This work is important to understand the fundamentals in networks where different RATs coexist [e.g., LTE (RAT-L) and WiFi (RAT-U)]. The authors derive the accurate void probability of the APs in each tier and they found the exact channel access probability for the opportunistic CSMA/CA with random backoff protocol. Relations for the coexisting coverage and network capacity are also derived.

In [53], the authors derive analytically, for a generic LAA eNB, the active probability and the coverage probabilities in the cases of single and multiple channels. The authors derive these probabilities in the random channel selection case and in the proposed channel selection mechanism.

In [23], the authors model the downlink coexistence between LTE-LAA system and the WiFi system using a Markov chain model. Then, they derive analytically the achieved average throughput on the downlink by both systems.

In [10], the authors study the effect on WiFi performance when LTE operates as a licensed user on the unlicensed spectrum. The results show a dramatic degradation in the WiFi throughput when LTE system has no LBT ability. Hence, the authors propose two coexistence methods that can be adopted by LTE on the unlicensed spectrum to provide fair coexistence with WiFi. The proposed mechanisms are non-coordinated (no collaboration happens between LTE and WiFi) and coordinated network managements. In the non-coordinated case, the authors propose an Adaptive LBT (ALBT) mechanism where the eNB senses all the channels and camps on one of the available idle channels and keeps switching between this pool of available channels which is being updated continuously. This prevents the eNB from excessively using one channel. In the coordinated case, the authors propose a system management method that combines LTE and WiFi systems through network function virtualization (NFV). This concept was supported by 3GPP and it allows different operators to share a pool of their allocated spectra with other operators. The authors suggest this concept to be used between LTE and WiFi on unlicensed band. According to the paper, this may eliminate the need for LBT as a whole and dealing with the co-channel interference.

6.2 System Model

The proposed system model consists of multiple cells deployed in the same geographical area but not necessarily the same coverage zone. These cells can be describe as follows

- Multiple LTE-eLAA cells from different operators.
- Multiple WiFi cells with dynamic channel allocation (DCA) ability.
- Multiple WiFi cells with no DCA ability (e.g., legacy cells).
- Out-of-System devices (OSDs) that operate on the unlicensed band not following the LBT rules (i.e., sources of interference on the channels).

The WiFi AP and its users use the CSMA/CA version of LBT. Meanwhile, the LTE-eLAA device accesses the medium using the basic LBT mechanism, which detects the

availability of the channel. If the channel is available, the cell accesses it without CTS/RTS handshake. Note that the control packets of the LTE-eLAA are sent through the licensed anchor frequency of the LTE-eLAA cell. This is unlike WiFi communication where the control packets are sent on the operating channel before the transmission happens.

The LTE-eLAA cell consists of an eNB and multiple UEs. Assume $e\mathcal{NB}$ be the set of all eNBs in the system and $\mathcal{UE}^{(i)}$ be the set of all UEs associated with $eNB^{(i)} \in e\mathcal{NB}$. The LTE-eLAA cell has downlink and uplink control data communications on a dedicated (anchor) licensed frequency.

The eNB and its UEs have modified LTE modules and MAC layer protocol that can operate according to the LTE standard over the unlicensed band. Note that the LTE eNB module has the ability to serve multiple UEs simultaneously on the same channel (each served UE has a different resource blocks/sub-channels within this channel).

The WiFi cell with DCA ability consists of an AP (AP_D) and multiple WiFi users (WU_D). Assume $\mathcal{AP}_\mathcal{D}$ be the set of all APs with DCA in the system and assume $\mathcal{WU}_\mathcal{D}{}^{(i)}$ be the set of all users associated with $AP_D^{(i)} \in \mathcal{AP}_\mathcal{D}$. These devices have WiFi modules with a modified WiFi MAC protocol capable to support the used DCA mechanism.

Note that the WiFi AP serves its users on a one-by-one bases on the chosen channel, unlike the eNB in LTE-eLAA where it serves the users simultaneously.

The WiFi cells without DCA ability are ordinary WiFi cells operating on the unlicensed band using the standard WiFi MAC protocol with no modifications. The cell consists of an AP (AP_{nD}) and multiple users (WU_{nD}s). Assume $\mathcal{AP}_{nD}$ be the set of all AP_{nD}s in the system and $\mathcal{WU}_{nD}^{(i)}$ be the set of all $WU_{nD}s$ associated with $AP_{nD}^{(i)} \in \mathcal{AP}_{nD}$.

6.2.1 System Technical Details And Assumptions

This section presents important details and assumptions of the proposed system model.

- We assume that if any device in the system granted the access to an unlicensed channel,

then it transmits on the maximum power (P_{max}) specified by the Federal Communications Commission (FCC). The OSDs transmit on a random power level from the interval $(0, P_{max})$.

- We assume the simplified pathloss model to model the radio propagation in the system.
- It is assumed that all the devices use an OFDM based modulation to gain the benefit of ignoring the adjacent channel interference.
- All the devices in the network are assumed to have half-duplex transceivers and an OFDM based modulation.
- We assume the use of $5GHz$ band to represent the unlicensed band. This band is divided into 25 channels with $20MHz$ bandwidth each.
- We assume, in each channel assignment, only one channel at most is used by any given cell. After finishing the transmission on the chosen channel, it must be released by the cell if DCA is used. From this we exclude the cells with no DCA, because they use only one channel (assumption) all the time for their operation.
- The used SINR-Data rate relation in LTE and WiFi networks is the staircase relation.
- The APs and eNBs are randomly deployed in the given geographical area.
- The WUs and UEs are randomly deployed inside their corresponding cells.
- Each cell has a traffic load value, $L_D^{(i)}$ for the i^{th} WiFi cell with DCA, $L_{nD}^{(i)}$ for the i^{th} WiFi cell with no DCA, and $L_e^{(i)}$ for the i^{th} LTE-eLAA cell.
- In a given time slot, there is a required data rates on the communication links between the AP or eNB and their corresponding users. Assume the data rate required on the link between $AP_D^{(i)}$ and its $WD_D^{(i,j)}$ be $R_D^{(i,j)}$, the data rate required between $AP_{nD}^{(i)}$ and its $WD_{nD}^{(i,j)}$ be $R_{nD}^{(i,j)}$, and the data rate required on the link between $eNB^{(i)}$ and its

$UE^{(i,j)}$ be $R_e^{(i,j)}$. It is possible for the required data rate to be zero, which means no transmission is needed at this time slot.

- To use EGTCA as a channel assignment mechanism, the AP_Ds, $eNBs$, WU_Ds, and UEs are required to sense the interference values on all the channels. The WU_Ds and UEs are required to communicate these values with their corresponding AP_Ds and $eNBs$ before the communication starts on their links (the MAC protocol details are left for future work). Assume the set of interference values sensed by $AP_D^{(i)}$ is given as $\mathcal{I}_D^{(i)} = [I_{D,1}^{(i)}, I_{D,2}^{(i)}, \ldots, I_{D,c}^{(i)}]$, where c is the total number of unlicensed channels in the system. Assume the set of interference values sensed by $WU_D^{(i,j)}$ is given as $\mathcal{I}_D^{(i,j)} = [I_{D,1}^{(i,j)}, I_{D,2}^{(i,j)}, \ldots, I_{D,c}^{(i,j)}]$. Assume the set of interference values sensed by $eNB^{(i)}$ on the same set of channels is given as $\mathcal{I}_e^{(i)} = [I_{e,1}^{(i)}, I_{e,2}^{(i)}, \ldots, I_{e,c}^{(i)}]$ and the set of interference values sensed by $UE^{(i,j)}$ on the same set of channels is given as $\mathcal{I}_e^{(i,j)} = [I_{e,1}^{(i,j)}, I_{e,2}^{(i,j)}, \ldots, I_{e,c}^{(i,j)}]$.

- It is assumed that different LTE-eLAA operators have no communication between them, this means that their communications are detectable only by using spectrum sensing. In the adopted system model, each operator is represented by only one LTE-eLAA cell.

6.3 Problem Formulations And Channel Assignment Mechanisms

Table 6.1 provides a list of the important symbols used in the problem formulation of the proposed channel assignment mechanisms (CAMs) in this section.

Symbol	Definition
M	Number of OSDs in the system
N_e	Number of eNBs in the system
N_D	Number of WiFi AP_Ds in the system
N_{nD}	Number of WiFi $AP_{nD}s$ in the system
$\mathcal{OSD} = \{OSD^{(1)}, OSD^{(2)}, \ldots, OSD^{(M)}\}$	Set of OSDs in the system
$e\mathcal{NB} = \{eNB^{(1)}, eNB^{(2)}, \ldots, eNB^{(N_e)}\}$	Set of eNBs in the system
$K_e^{(i)}$	Number of UEs associated with $eNB^{(i)}$
$\mathcal{UE}^{(i)} = \{UE^{(i,1)}, UE^{(i,2)}, \ldots, UE^{(i,K_e^{(i)})}\}$	Set of UEs associated with $eNB^{(i)}$
$\mathcal{AP}_D = \{AP_D^{(1)}, \ldots, AP_D^{(N_D)}\}$	Set of AP_Ds in the system
$K_D^{(i)}$	Number of WU_Ds associated with $AP_D^{(i)}$
$\mathcal{WU}_D^{(i)} = \{WU_D^{(i,1)}, WU_D^{(i,2)}, \ldots, WU_D^{(i,K_D^{(i)})}\}$	Set of WU_Ds associated with $AP_D^{(i)}$
$\mathcal{AP}_{nD} = \{AP_{nD}^{(1)}, \ldots, AP_{nD}^{(N_{nD})}\}$	Set of $AP_{nD}s$ in the system
$K_{nD}^{(i)}$	Number of WU_{nD}s associated with $AP_{nD}^{(i)}$
$\mathcal{WU}_{nD}^{(i)} = \{WU_{nD}^{(i,1)}, WU_{nD}^{(i,2)}, \ldots, WU_{nD}^{(i,K_{nD}^{(i)})}\}$	Set of $WU_{nD}s$ associated with $AP_{nD}^{(i)}$
c	Total number of unlicensed channels in the system
$\mathcal{C} = \{ch_1, ch_2, \ldots, ch_c\}$	The set of all unlicensed channels in the system
$L_e^{(i)}$	Traffic load of $eNB^{(i)}$ cell
$L_D^{(i)}$	Traffic load of $AP_D^{(i)}$ cell
$L_{nD}^{(i)}$	Traffic load of $AP_{nD}^{(i)}$ cell
$\mathcal{I}_e^{(i)} = \{I_{e,1}^{(i)}, I_{e,2}^{(i)}, \ldots, I_{e,c}^{(i)}\}$	Set of interference values on each channel sensed by $eNB^{(i)}$
$\mathcal{I}_e^{(i,j)} = \{I_{e,1}^{(i,j)}, I_{e,2}^{(i,j)}, \ldots, I_{e,c}^{(i,j)}\}$	Set of interference values on each channel sensed by $UE^{(i,j)}$
$\mathcal{I}_D^{(i)} = \{I_{D,1}^{(i)}, I_{D,2}^{(i)}, \ldots, I_{D,c}^{(i)}\}$	Set of interference values on each channel sensed by $AP_D^{(i)}$
$\mathcal{I}_D^{(i,j)} = \{I_{D,1}^{(i,j)}, I_{D,2}^{(i,j)}, \ldots, I_{D,c}^{(i,j)}\}$	Set of interference values on each channel sensed by $WU_D^{(i,j)}$
$\mathcal{I}_{nD}^{(i)} = \{I_{nD,1}^{(i)}, I_{nD,2}^{(i)}, \ldots, I_{nD,c}^{(i)}\}$	Set of interference values on each channel sensed by $AP_{nD}^{(i)}$
$\mathcal{I}_{nD}^{(i,j)} = \{I_{nD,1}^{(i,j)}, I_{nD,2}^{(i,j)}, \ldots, I_{nD,c}^{(i,j)}\}$	Set of interference values on each channel sensed by $WU_{nD}^{(i,j)}$
$\mathcal{R}_e^{(i)} = \{R_e^{(i,1)}, R_e^{(i,2)}, \ldots, R_e^{(i,K_e^{(i)})}\}$	Set of required rates on each link between $eNB^{(i)}$ and its UEs
$\mathcal{R}_D^{(i)} = \{R_D^{(i,1)}, R_D^{(i,2)}, \ldots, R_D^{(i,K_D^{(i)})}\}$	Set of required rates on each link between $AP_D^{(i)}$ and its WU_Ds
$\mathcal{R}_{nD}^{(i)} = \{R_{nD}^{(i,1)}, R_{nD}^{(i,2)}, \ldots, R_{nD}^{(i,K_{nD}^{(i)})}\}$	Set of required rates on each link between $AP_{nD}^{(i)}$ and its $WU_{nD}s$

Table 6.1. Notations summary

Next, we propose three must-have important aspects (Strength Aspects) for any CAM to be efficient

1. An effective objective function.

2. The ability to achieve the objective function.

3. The ability to account for the current surrounding activities of the network.

In this section we are considering five types of CAMs. In the following we list these CAMs along with the corresponding problem formulations if needed.

The Random CAM:

This CAM is used, throughout this work, only by WiFi cells with no DCA ability. Here, at the time of AP deployment, the AP chooses randomly a channel and stays on it for a long period of time (we can assume this time to be infinity). This CAM is the same as the one used in Chapter 5 and in [47]. Note that this CAM does not possess any of the strength aspects. It is worth mentioning that the Random CAM is vastly used in real-life WiFi APs along with the Greedy CAM presented later on. The reason to make this CAM exclusive to WiFi cells with no DCA ability is to have the results as close as possible to a real-life scenario.

The Enhanced-Random CAM:

This CAM can be used only, throughout this work, by AP_D and eNB cells (i.e., WiFi with DCA ability and eLAA networks, respectively). Here, the AP_D or eNB opens an Access Window (AW) every time it receives a request for transmission. In this AW, all the needed transmissions in the cell are shared with the AP_D or eNB. After this, the AP_D or eNB chooses a channel randomly. If the channel is currently available and can support the needed rate on a given link, then the cell can have successful transmission on this link. If the channel is currently occupied or cannot support the needed rate, then the transmission on this link gets blocked. After processing (i.e., allowing or blocking) each of the needed transmissions declared in the AW, the cell must release the channel. Note that this channel assignment does not have any of the strength aspects but up to a certain level it accounts for network activity because

1) it releases the channel after each transmission, this will give the chance for other cells to use the released channel which may rise the network throughput.

2) it chooses a new channel randomly the per-AW basis. This raises the chance for the cell itself to have higher throughput values since it keeps hopping between channels other

than staying on one channel, which may not be good at all, as in the Random CAM.

Hence, we expect better results using the Enhanced-Random CAM compared to the Random-CAM.

The Greedy CAM:

This CAM can be used only by eNB cells an WiFi cells with DCA ability. Here, at the time of eNB/AP deployment, the eNB/AP chooses the channel with the lowest interference plus noise (IN) value and stays on it for a long period of time (we can assume this time to be infinity). This CAM is the same as the one used in Chapter 5 and in [47]. Note that this CAM possesses only the first strength aspect. As a result, this CAM can waste a lot of throughput as was shown in Chapter 5 and in [47]. This is because all the cells will choose the "best" channels and leave the opportunities available in the "bad" channels. Having too many cells using one good channel will result in high contention, and hence, high blocking rate. It is worth mentioning that the Greedy CAM is also vastly used in real-life WiFi APs.

The reason we do not apply the Greedy CAM to the "WiFi cells with no DCA ability" is that "WiFi cells with DCA ability" using the Greedy CAM is the same as "WiFi cells with no DCA ability" using the Greedy CAM. Hence, for realistic results, it is good to have WiFi cells using the Random CAM coexisting with other WiFi cells using the Greedy CAM.

The Enhanced-Greedy CAM:

This CAM can be used only by AP_D and eNB cells. Here, the AP_D or eNB opens an AW every time it receives a request for transmission. In this AW, all the needed transmissions in the cell are shared with the AP_D or eNB. After this, the AP_D or eNB chooses the best channel (i.e., the channel with the least IN value at the AP/eNB). If this channel is currently occupied, it chooses the second best channel, if it is also occupied, it chooses the third best channel and so on. After processing (i.e., allowing or blocking) each of the needed transmissions declared in the AW, the cell must release the channel. Note that this channel assignment has all of the three strength aspects. Hence, it is expected to have much better results compared to all of the previously mentioned CAMs. This channel assignment has a

credible objective function (i.e., Strength Aspect 1), it is able to satisfy the objective function by choosing the best currently available channel (i.e., Strength Aspect 2), and accounts for the current network activities by choosing the channel that is currently available and releases the channel after finishing the transmission (i.e., Strength Aspect 3).

The Enhanced Generous Throughput Oriented CAM (EGTCA):

This CAM can be used only by AP_D and eNB cells. Here, the AP_D or eNB opens an AW every time it receives a request for transmission. In this AW, all the needed transmissions in the cell are shared with the AP_D or eNB. After this, the AP_D or eNB chooses the channel with the highest EGTCA objective function value (OFV) (described next). If this channel is currently occupied, it chooses the second highest EGTCA OFV channel, if it is also occupied, it chooses the third highest EGTCA OFV channel and so on. After processing (i.e., allowing or blocking) each of the needed transmissions declared in the AW, the cell must release the channel.

This CAM is based on the GTCA method proposed in Chapter 5 and in [47]. The main objective of this CAM is for the AP_D and eNB cells to use the channel that "is just good enough" for communication and leave the over qualified channels to other cells that may need them. Here,"how good the channel is" is defined by the IN value seen on it by the cell in a similar sense as in GTCA but the algorithm is simpler to implement and less conservative compared to the original GTCA. Less conservative means that the EGTCA allows neighboring cells to share the same channel, where GTCA prioritizes choosing a channel from the set of unused channels by the neighboring cells.

One feature that someone would think of is to use the channel load in the channel assignment mechanism and define the "just good enough channel" as the channel with the highest IN value and highest load but still satisfies the requirements of the cell. In our work, adding the channel load to the channel assignment mechanism is not possible for the following reasons

- There are three required objectives: 1) maximize the number of served users, 2) choos-

ing the channel with the maximum IN value, and 3) choosing the channel with the highest load value. The most important objective is maximizing the number of served users. Now, it is hard to decide which is more important than the other between the second and the third objectives (i.e. do we prefer the channel with the highest interference or the channel with the highest load if they both support the same number of users?).

- EGTCA is implemented per AW (i.e., after each AW, the AP/eNB runs EGTCA). This means the network is highly dynamic and no need for the cell to stay on the channel for an extended period of time. If the cell needs to occupy the channel for a long period of time, then the channel load matters, but if the cell is making the decisions per AW, then what matters is the instantaneous occupancy of the channel and how fast can the cell deliver the needed files (i.e., data).

- If in the same vicinity one channel has high IN value but satisfies the needed rates by a number of cells, then it will be preferable for all of them. Hence, this channel will experience high traffic load automatically without the need to integrate the channel load in the assignment process.

- In EGTCA, it is more important to choose the channel with the highest IN value than the channel with the highest load. This is because we need the cell to transmit with the minimal bandwidth (BW) resources and minimal time resources. Hence, choosing the channel with the highest load means that the cell has the highest chances to get blocked because of the high activity on the channel, which means more blocking and more time resources being used.

- The channel load is important to consider, if the cell is using the same channel for a long time (not per AW), but since the cell is using the channel for the current requirements and then leaves the channel, then there is no need to consider the channel load and what is important is the channel being currently available or not.

It is worth mentioning, that the EGTCA and GTCA have the three Strength Aspects presented preciously. These channel assignments have a credible objective functions (i.e., Strength Aspect 1), they are able to satisfy the objective function by choosing the "just good enough" and currently available channel (i.e., Strength Aspect 2), and accounts for the current network activities by choosing the channel that is currently available and releases the channel after finishing the transmission (i.e., Strength Aspect 3).

In Chapter 5 and [47], the fact that GTCA satisfies all the Strength Aspects is a more solid explanation behind the tremendous GTCA throughput gain compared to the Greedy and Random CAMs. This explanation was harder to recognize, at first, in the presented work in Chapter 5 and in [47]. Being "generous" is one reason behind this astonishing performance, but the other reasons are the accounting for the surrounding activities, and releasing the channel after processing the transmissions. Being "generous" is Strength Aspect 1, the ability to find a channel that satisfies the objective function is Strength Aspect 2, and the ability to account for the current network activity is Strength Aspect 3. Strength Aspect 3 is resembled by: 1) leaving the chosen channel after transmission, and 2) choosing the highest objective-function valued channel that satisfies the objective function **and currently available**. For example, if the highest objective-function valued channel is currently occupied, this will result in a total blocking of the transmission. Hence, choosing the second highest channel, but currently available has higher probability in transmitting the data either partially or totally.

Based on this, we propose next the EGTCA version for eNBs and APs with DCA ability.

Problem formulation at $eNB^{(i)}$

As mentioned previously the objective at $eNB^{(i)}$ is to find the worst channel that maximizes the number of served users by $eNB^{(i)}$. The worst channel means the currently available channel with the highest IN value seen by $eNB^{(i)}$. According to this, define $X_e^{(i,j)}$ as:

$$X_e^{(i,j)} = \begin{cases} 1, & \text{if } UE^{(i,j)} \text{ can be supported by } eNB^{(i)} \text{ on the chosen channel } ch_e^{(i)} \\ 0, & \text{otherwise} \end{cases}. \quad (6.1)$$

In LTE, since the users, which are allowed to transmit, use the channel (with 20 MHz bandwidth as assumed before) simultaneously, then for the sake of fairness, the eNB will distribute this bandwidth equally between the users (not all the users in the eNB cell, but only those which are approved to transmit by the eNB). Now, by assuming $d \in \{1, \ldots, K_e^{(i)}\}$ be the possible number of portions (i.e. divisions or sub-channels) of the chosen channel, then the problem formulation at $eNB^{(i)}$ can be given as follows:

$$\max_{ch_e^{(i)} \in \mathcal{C}, d} \Big\{ A \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} + \sum_{q=1}^{c} I_{e,q}^{(i)} v_q \Big\} \quad (6.2)$$

s.t.

- $v_q = \begin{cases} 1, & \text{if } ch_e^{(i)} = ch_q \text{ and } ch_q \text{ is currently available} \\ 0, & \text{otherwise} \end{cases}$
- $\sum_{q=1}^{c} v_q \leq 1$; (exclusivity/blocking constraint)
- $X_e^{(i,j)} = \begin{cases} 0, & \text{if } R_e^{(i,j)} > \frac{1}{d} R_e\big(\min\big(SINR_{\text{Down}}^{(i,j,q)}, SINR_{\text{Up}}^{(i,j,q)}\big)\big) \\ 1, & \text{otherwise} \end{cases}, \forall j = 1, \ldots, K_e^{(i)}$
- $d = \begin{cases} \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)}, & \text{if } \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} \geq 1 \\ 1, & \text{otherwise (i.e. } \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} = 0) \end{cases}$

Where $A = \max(\mathcal{I}_e^{(i)})$, $R_e(SINR)$ is the LTE-eLAA SINR-data rate, $SINR_{\text{Down}}^{(i,j,q)}$ is the SINR at $UE^{(i,j)}$ from $eNB^{(i)}$ on channel ch_q, and $SINR_{\text{Up}}^{(i,j,q)}$ is the SINR at $eNB^{(i)}$ from $UE^{(i,j)}$ on channel ch_q.

Note that the third constraint is formed in this way because of the simultaneous transmissions of the LTE users on the chosen channel. A is used to give the priority to the channel that maximizes the number of UEs, if there are more than one channel that gives the same number, then we choose the one with the highest interference. The last constraint is used so that the number of served users is less than or equal to the number of sub-channel (i.e. channel divisions). It is important to mention that $d \neq \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)}$ all the time since there is a possible case where no UE can be served by $eNB^{(i)}$ (i.e. $\sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} = 0$) while $d = 1$ in this case.

The optimization problem at $eNB^{(i)}$ consists of two optimization problems: 1) find the maximum number that can be supported on each available channel, and 2) find the channel that has the maximum supported number of users among all the channels. According to this, the optimization problem in (6.2) can be solved optimally using Algorithm 1.

The fitness function value on a given channel ch_q (FFV_q) is: $FFV_q = A \times MUN_q + I_{e,q}^{(i)}$. Where MUN_q is the maximum number of users that can be supported if $eNB^{(i)}$ chooses channel ch_q for transmission.

Note that eNB cell does not give any priority to the users with high required data rates. This is for the sake of fairness in the cell, all users with different required data rates should be considered with the same priority by the eNB.

Algorithm 3 $eNB^{(i)}$ EGTCA channel assignment pseudo code

1: **Inputs:**

2: $R_e^{(i)}$; the pairs $(SINR_{Down}^{(i,j,q)}, SINR_{Up}^{(i,j,q)}), \forall j = 1, \ldots, K_e^{(i)},\ q = 1, \ldots, c$

3: **Initialization:**

4: Let $Ch_e^{(i)} \leftarrow Void$

5: **procedure** Channel assignment

6: **for** ($q \leftarrow 1$ to c) **do**

7: $MUN\_Old = 0$

8: **for** ($d \leftarrow K_e^{(i)}$ to 1) **do**

9: **for** ($j \leftarrow 1$ to $K_e^{(i)}$) **do**

10: **if** $\left(R_e^{(i,j)} > \frac{1}{d} R_e\big(\min(SINR_{\text{Down}}^{(i,j,q)}, SINR_{\text{Up}}^{(i,j,q)})\big)\right)$ **then**

11: $X_e^{(i,j)} = 0$

12: **else**

13: $X_e^{(i,j)} = 1$

14: **end if**

15: **end for**

16: **if** $\sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} > d$ **then**

17: Pick randomly d users from all the users with $X_e^{(i,j)} = 1$, then force the rest of $X_e^{(i,j)}$s to take the value of 0.

18: **else if** $0 < \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)} \leq d$ **then**

19: Force all $X_e^{(i,j)}$ to equal zero.

20: **end if**

21: $MUN = \sum_{j=1}^{K_e^{(i)}} X_e^{(i,j)}$

22: **if** ($MUN > MUN\_Old$) **then**

23: $MUN\_Old \leftarrow MUN$

24: **end if**

25: **if** ($MUN\_Old == K_e^{(i)}$) **then**

26: Break out of the For loop.

27: **end if**

28: **end for**

29: $MUN \leftarrow MUN\_Old$

30: $FFV_q = A \times MUN + I_{e,q}^{(i)}$

31: **end for**

32: Arrange the channels according to their FFVs.

33: $Ch_e^{(i)} \leftarrow$ The channel with the highest FFV and currently available. If not, go to the next step.

34: $Ch_e^{(i)} \leftarrow$ The channel with the second highest FFV and currently available. If not, go to the next step. And so on.

35: If non of the channels are available, block all the needed transmission and $Ch_e^{(i)} \leftarrow Void$.

36: **end procedure**

Line 17 in the algorithm is important to satisfy the last constraint in (6.2).

Problem formulation at $AP_D^{(i)}$

Here, $AP_D^{(i)}$'s cell opens a AW with a number of time slots. In each time slot a WU_D is allowed to share its transmission information with $AP_D^{(i)}$. Then $AP_D^{(i)}$ chooses the channel that maximizes the number of served users (by $AP_D^{(i)}$). This channel should be the one with the highest interference value but still can serve the maximum possible number of WU_Ds while being currently available. Hence, the problem formulation can be given as follows:

$$\max_{ch_D^{(i)} \in \mathcal{C}} \left\{ A \sum_{j=1}^{K_D^{(i)}} X_D^{(i,j)} + I_{D,q}^{(i)} \right\} \tag{6.3}$$

s.t.

- $v_q = \begin{cases} 1, & \text{if } ch_D^{(i)} = ch_q \text{ and } ch_q \text{ is currently available} \\ 0, & \text{otherwise} \end{cases}$
- $\sum_{q=1}^{c} v_q \leq 1$; (exclusivity/blocking constraint)
- $X_D^{(i,j)} = \begin{cases} 1, & \text{if } R_D^{(i,j)} \leq \sum_{q=1}^{c} v_q \times R_D(\min\left(SINR_{\text{Down}}^{(i,j,q)}, SINR_{\text{Up}}^{(i,j,q)}\right)) \\ 0, & \text{otherwise} \end{cases}$.

Where $ch_D^{(i)}$ is the chosen channel at $AP_D^{(i)}$, $R_D(SINR)$ is the WiFi SINR-data rate relation, and $A = \max(\mathcal{I}_D^{(i)})$. Note that the last constraint is formulated in this way to comply with the property of the WiFi AP serving the users on the chosen channel on a one-by-one basis. This is unlike the LTE eNB that can serve multiple uses on the chosen channel simultaneously. Note that the optimization problem at $AP_D^{(i)}$ consists of two optimization problems 1) find the maximum number that can be supported on each available channel, and 2) find the channel that has the maximum supported number of users among all the channels. According to this, the optimization problem in (6.3) can be solved optimally using Algorithm 2.

Algorithm 4 $AP_D^{(i)}$ EGTCA channel assignment pseudo code

Inputs: $\mathcal{R}_D^{(i)}$

Find the raw vector $SINR_{Links}^{(i,q)} = \big(\min(SINR_{\text{Down}}^{(i,j,q)}, SINR_{\text{Up}}^{(i,j,q)})\big) \in \mathbb{R}^{1\times K_D^{(i)}}, \forall q = 1, ..., c.$

Let $Ch_D^{(i)} \leftarrow Void$

procedure CHANNEL ASSIGNMENT

  for ($q \leftarrow 1$ to c) **do**

    $MUN\_Old = 0$

    for ($j \leftarrow 1$ to $K_D^{(i)}$) **do**

      if $R_D^{(i,j)} \leq R_D\big(SINR_{Links}^{(i,q)}(j)\big)$ **then**

        then $X_D^{(i,j)} = 1$

        else $X_D^{(i,j)} = 0$

      end if

    end for

    $MUN = \text{sum}(X_D^{(i,j)} \forall j)$

    $FFV_q = A \times MUN + I_{e,q}^{(i)}$

  end for

  Arrange the channels according to their FFVs.

  $Ch_D^{(i)} \leftarrow$ The channel with the highest FFV and currently available. If not, go to the next step.

  $Ch_D^{(i)} \leftarrow$ The channel with the second highest FFV and currently available. If not, go to the next step. And so on.

  If non of the channels are available, block all the needed transmission and $Ch_D^{(i)} \leftarrow Void$.

end procedure

Note that the eNBs and APs with DCA ability can be equipped with DCA mechanism (such as EGTCA, Enhanced-Greedy, and Enhanced-Random). They can also be equipped with non-DCA mechanisms such as (Random CAM and the Greedy CAM). Regarding the APs with no DCA ability, they can only be equipped with a non-DCA mechanism such as the (Random CAM and the Greedy CAM).

Note that in the proposed network model, there is no overall system optimization and all the optimization is done locally at the cell (eNB or AP_D cell) level. Even though the over all system optimization might provide better throughput results, but on the other hand,

it is impractical since the cells are owned by different operators and administrators. This makes the sharing of the needed optimization information between the cells very restricted or impossible in real life scenarios.

6.3.1 Fairness Calculations

This section presents the fairness calculation relations on a specific network level (i.e., LTE network level, or WiFi with DCA ability network level, or WiFi without DCA ability network level), or among all networks. These relations are based on Jain's fairness index (FI) in [43].

Fairness on LTE network level:

$$FI_e = \frac{(\sum_{j=1}^{N_e} x_j)^2}{\sum_{j=1}^{N_e} x_j^2},$$

where x_j is the total throughput achieved by the j^{th} LTE-eLAA cell.

Fairness on WiFi with DCA network level:

$$FI_D = \frac{(\sum_{j=1}^{N_e} x_j)^2}{\sum_{j=1}^{N_D} x_j^2},$$

where x_j is the total throughput achieved by the j^{th} AP_D cell.

Fairness on WiFi without DCA network level:

$$FI_{nD} = \frac{(\sum_{j=1}^{N_e} x_j)^2}{\sum_{j=1}^{N_{nD}} x_j^2},$$

where x_j is the total throughput achieved by the j^{th} AP_{nD} cell.

Fairness among all networks:

Since the networks differ in the way they achieve their throughput, we propose the idea of replacing the throughput value by the number of channel accesses per cell in the Jain's fairness index relation to calculate the fairness among all the coexisting networks. Based on this, we have the following fairness index value for all the networks

$$FI_{\text{All}} = \frac{(\sum_{i=1}^{N_e} x_i + \sum_{j=1}^{N_D} y_j + \sum_{k=1}^{N_{nD}} z_k)^2}{\sum_{i=1}^{N_e} x_i^2 + \sum_{j=1}^{N_D} y_j^2 + \sum_{k=1}^{N_{nD}} z_k^2},$$

where x_i is the total number of spectrum accesses by the i^{th} eNB cell. y_j is the total number of spectrum accesses by the j^{th} AP_D cell. z_k is the total number of spectrum accesses by the k^{th} AP_{nD} cell.

6.4 Performance Evaluation

In this section, the performance of the proposed CAMs (i.e., EGTCA, and Enhanced-Greedy) is tested against the performance of the Greedy, the Enhanced-Random, and the Random CAMs. The interactions between different networks (i.e., heterogeneous networks) is also studied through two main metrics: the achieved network throughput, and network fairness.

6.4.1 Simulation Setup

The simulation setup consists of three types of networks: 1)LTE-eLAA, 2) WiFi with DCA ability, and 3) WiFi with no DCA ability. The system also consists of OSDs acting as sources of interference. In this section, there are four system setups as in Table 6.2.

Setup \ Network	LTE-eLAA	WiFi with DCA ability	WiFi with no DCA ability
EGTCA Setup	EGTCA CAM	EGTCA CAM	Random CAM
E-Greedy Setup	Enhanced-Greedy CAM	Enhanced-Greedy CAM	Random CAM
Greedy Setup	Greedy CAM	Greedy CAM	Random CAM
E-Random Setup	Enhanced-Random CAM	Enhanced-Random CAM	Random CAM

Table 6.2. System Setups

From Table 6.2, the first setup is "EGTCA". In this setup, the eLAA network uses EGTCA CAM, the WiFi network with DCA ability also uses the EGTCA CAM, while the WiFi network with no DCA ability uses the Random CAM. In the E-Greedy setup, the eLAA

and WiFi with DCA ability networks use Enhanced-Greedy CAM, while the WiFi with no DCA ability uses the Random CAM. In the Greedy setup, the eLAA and WiFi with DCA ability networks use Greedy CAM, while the WiFi with no DCA ability uses the Random CAM. Lastly, in the E-Random setup, the eLAA and WiFi with DCA ability networks use the Enhanced-Random CAM, while the WiFi with no DCA ability uses the Random CAM.

Note that the WiFi with no DCA ability network always uses the Random CAM. This is very important in making the simulations more realistic (since this is the mostly used WiFi network in real life scenarios), and to study the performance in the existence of such network. The other network that is also widely used in real life is the WiFi with no DCA ability network that uses the Greedy CAM. This type is also considered in our simulations in the third setup (i.e., the Greedy setup), where the WiFi with DCA ability network using the Greedy CAM acts exactly as WiFi with no DCA ability network using the Greedy CAM.

It is worth mentioning that in all the simulations, we adopted the SINR-Data rate relation of IEEE 802.11n given in Table 6.3. All types on networks (i.e., eLAA, WiFi with DCA ability, and WiFi with no DCA ability) refer to this table to decide the data rate given the SINR on a given link. Unifying all networks by the same SINR-Data rate relation is due to two reasons: 1) to test the coexistence performance while isolating the differences in the SINR-Data rate relations, and 2) regarding the LTE-eLAA, there is no specific way to calculate the data rate based on the SINR value since the relation differs for different manufacturers. Hence, for the sake of fairness and simplicity, we unified all the networks by having the same SINR-Data rate relation.

SINR (dB)	< 2	2-5	5-9	9-11	11-15	15-18	18-20	20-25	$25 <$
Data Rate (Mbps)	0	6.5	13	19.5	26	39	52	58.5	65

Table 6.3. IEEE 802.11n SINR-Data rate relation.

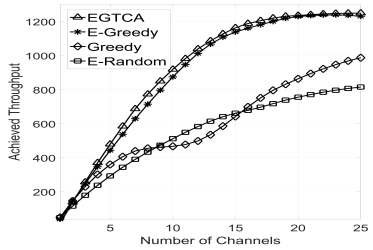
In all the simulations, we assume the path-loss exponent $\alpha = 3.5$, the average noise

power -84dBm, the maximum transmit power $P_{max} = 23$dBm, and the transmitter and receiver gains are 1dB each.

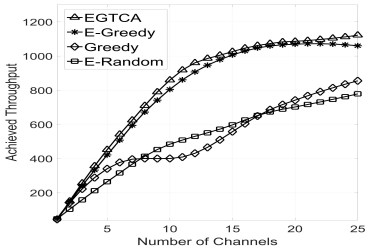
6.4.2 Simulations

Simulation 1: This simulation consists of $N_e = 5$ eLAA cells, $N_D = 5$ WiFi with DCA ability cells, and $N_{nD} = 5$ WiFi with no DCA ability cells. These cells are randomly deployed in a 50m$\times$50m area. Each cell has at maximum 10 users also randomly deployed in their corresponding cells. The traffic load of each cell is chosen randomly from interval $(0, 0.1)$, this means that a cell with traffic load $l \in (0, 0.1)$ has a probability of l to have a transmission request at any given time. In this simulation, if a user transmission is approved by the cell, then the number of transmitted packets per user is $P_{Num} = 10$ with each packet has the size of 1K Byte. The transmissions happen on the 5GHz band which is divided into 25 channels, each with 20MHz bandwidth. To calculate the averaged achieved throughput by each network, the total successfully transmitted data by the network is averaged over the total simulation time and P_{Num}. If a cell user has a transmission, then the required rate by this user is chosen randomly from the set $\{6.5, 13, 19.5, 26, 39, 52, 58.5, 65\}$. In this simulation, the number of OSDs is $M = 25$, where each OSD transmits on a power level randomly chosen from the interval $(0, P_{max})$. These OSDs are randomly deployed through the area.

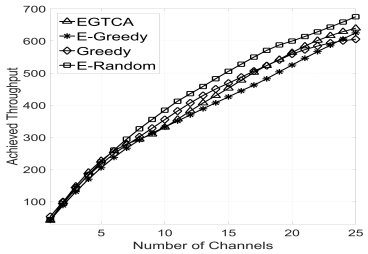
In this simulation we study the throughput and fairness performance as the number of channels used in the system (i.e., c) increases.



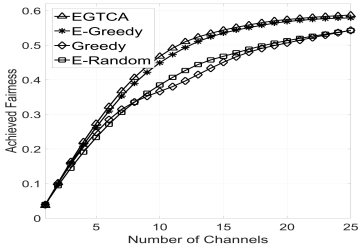
(a) eLAA Throughput (Mbps) Vs c.



(b) WiFi with DCA throughput (Mbps) Vs c.



(c) WiFi without DCA throughput (Mbps) Vs c.



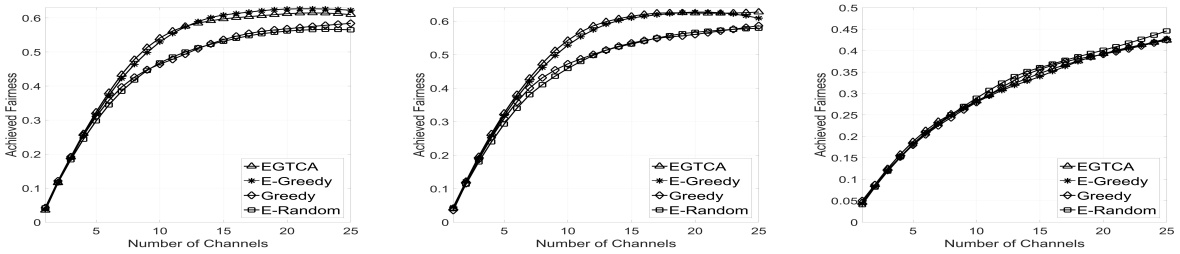
(d) FI_{All} Vs c.

Figures 6.1(a), 6.1(b), and 6.1(c) plot the achieved throughput as c increases under the four setups for the networks eLAA, WiFi with DCA ability, and WiFi with no DCA ability, respectively. As it is clear from Figures 6.1(a) and 6.1(b) that the EGTCA and E-Greedy setups outperforms the Greedy and E-Random setups significantly, this is because of the use of the EGTCA CAM and Enhanced-Greedy CAM in the EGTCA setup and E-Greedy setup, respectively. This outperformance is due to the fact that both the EGTCA and Enhanced-Greedy satisfy the previously mentioned strength aspects while the Greedy and Random CAMs do not satisfy such strength aspects. We also notice that the EGTCA setup slightly outperforms the E-Greedy setup since the objective function used by the EGTCA CAM is more efficient than the objective function in the Enhanced-Greedy CAM.

We also notice a strange throughput behavior of the Greedy setup where the throughput increases as c increases, then a state of "temporary saturation" happens, and then the throughput start to increase again as c increases. We call this"temporary saturation": the "piling effect". When the number of channels is very small, then there is a high probability that all the channels are contaminated with interference (let us call this set of channels the

"bad set"), hence all cells in the network compete on these channels and make use of the transmission opportunities available in them. Note that this set of "bad" channels differs from one cell to another based on the cell location with respect to the interference sources. This means the total network load is distributed over all the channels in the "bad set". As c increases, the chance of getting an interference-free channels gets larger, hence when a set of interference-free channels is available (let us call this set the "good set") all the cells in the network choose their operating frequency from this "good set" and ignore all the transmission opportunities available in the "bad set" (according the Greedy CAM). This creates congestion (i.e., piling) on the good set which results in this temporary saturation. As c increases, the size of the "good set" increases which means a decrease in the congestion, hence higher throughput values are possible.

Regarding the total fairness, we notice that both the EGTCA and E-Greedy setups outperform the other setups because of satisfying the second and third strength aspects. We also notice that the EGTCA and E-Greedy have very close fairness behavior.



(a) eLAA fairness Vs c.

(b) WiFi with DCA fairness Vs c.

(c) WiFi without DCA fairness Vs c.

Figure 6.2. Fairness of individual networks under different setups.

From Figures 6.2(a) and 6.2(b), we can see that EGTCA and E-Greedy setups provide approximately similar fairness performance for eLAA and WiFi with DCA ability networks, respectively. As expected, the EGTCA and E-Greedy outperform both the Greedy and E-Random Setups for the eLAA and WiFi with DCA ability networks.

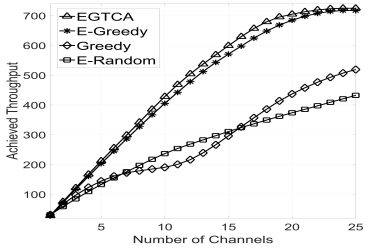
Regarding the WiFi without DCA ability, this network has very close fairness performance under all the network setups as clear from Figure 6.2(c).

From Figures 6.1(c) and 6.2(c), we can see that having new types of networks such as eLAA and WiFi with DCA ability operating on the same band with already existing WiFi network (i.e., WiFi without DCA ability) will have a negligible effect on the latter network. Hence, the coexistence is possible under any system setup since the WiFi without DCA ability network wastes a huge amount of transmission opportunities which can be used by the newly added networks. Wasting this amount of transmission opportunities is due to the fact that the Random CAM does not possess any of the strength aspects.

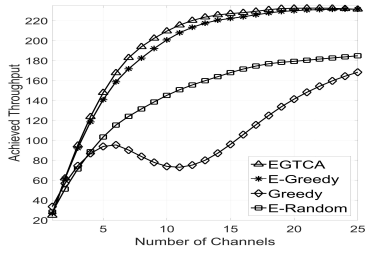
Simulation 2: This simulation has the same system properties as in Simulation 1 but with $P_{Num} = 20$ packets. This simulation studies each network behavior if the users of all the cells are forced to ask for the same data rate (let us call it R^*) for their transmissions. In this simulation the value of R^* can take one of the values from the set $\{6.5, 13, 19.5, 26, 39, 52, 58.5, 65\}$.

The Figures from 6.3(a) to 6.10(c) show the achieved throughput by each network for each possible R^* value.

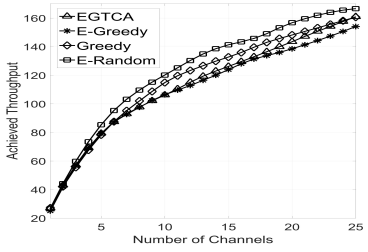
- When $R^* = 6.5$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



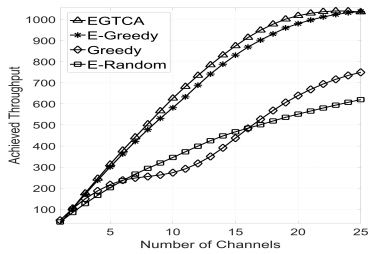
(b) WiFi with DCA throughput (Mbps) Vs c.



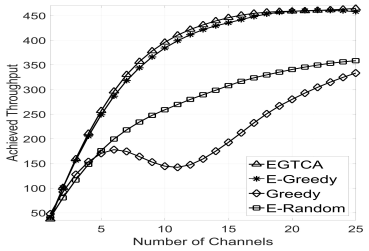
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.3. $R^* = 6.5$Mbps.

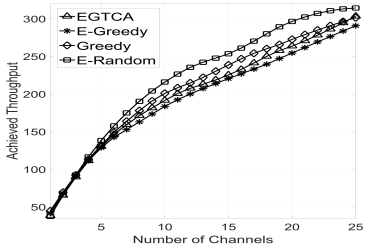
- When $R^* = 13$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



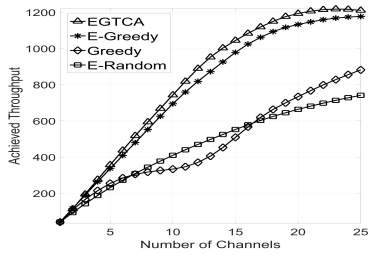
(b) WiFi with DCA throughput (Mbps) Vs c.



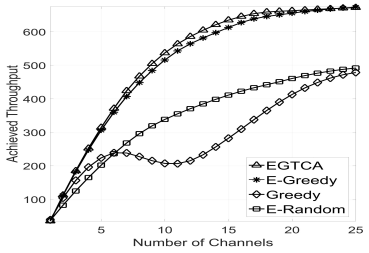
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.4. $R^* = 13$Mbps.

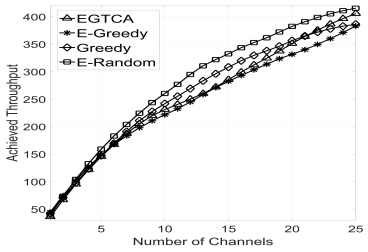
- When $R^* = 19.5$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



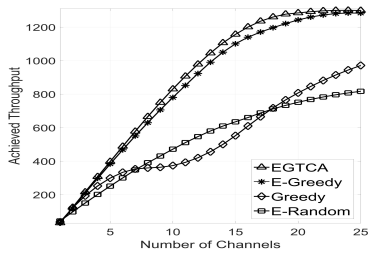
(b) WiFi with DCA throughput (Mbps) Vs c.



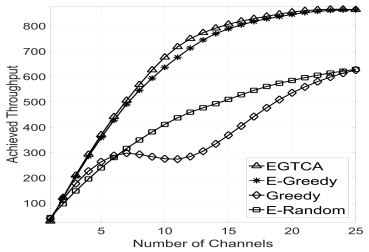
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.5. $R^* = 19.5$Mbps.

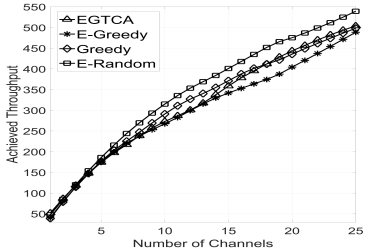
- When $R^* = 26$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



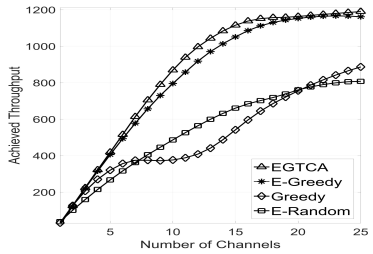
(b) WiFi with DCA throughput (Mbps) Vs c.



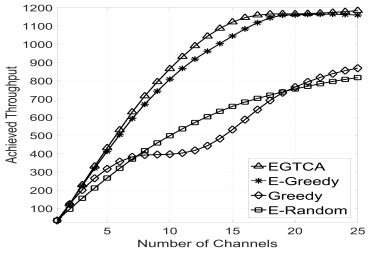
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.6. $R^* = 26$Mbps.

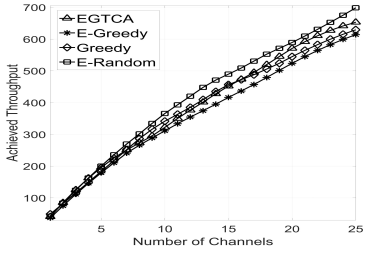
- When $R^* = 39$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



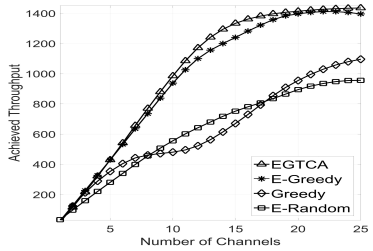
(b) WiFi with DCA throughput (Mbps) Vs c.



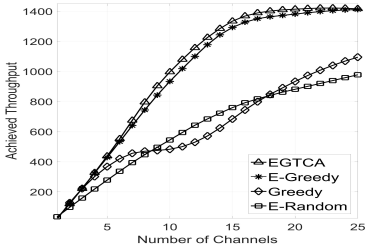
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.7. $R^* = 39$Mbps.

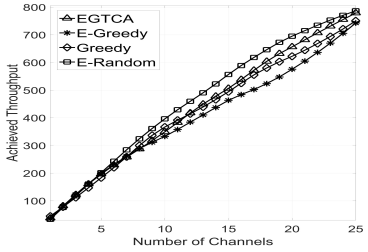
- When $R^* = 52$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



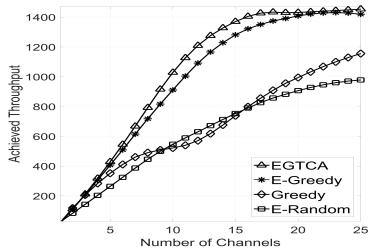
(b) WiFi with DCA throughput (Mbps) Vs c.



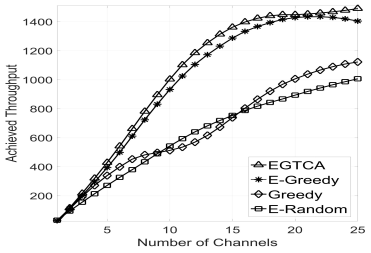
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.8. $R^* = 52$Mbps.

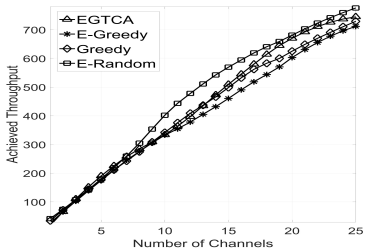
- When $R^* = 58.5$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



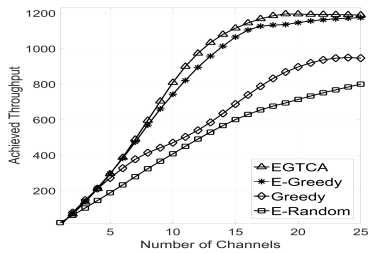
(b) WiFi with DCA throughput (Mbps) Vs c.



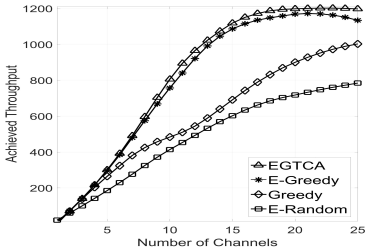
(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.9. $R^* = 58.5$Mbps.

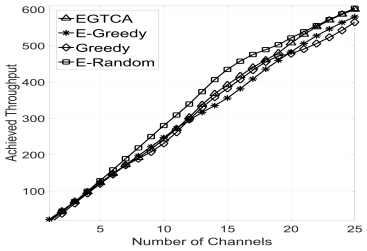
- When $R^* = 65$Mbps:



(a) eLAA Throughput (Mbps) Vs c.



(b) WiFi with DCA throughput (Mbps) Vs c.



(c) WiFi without DCA throughput (Mbps) Vs c.

Figure 6.10. $R^* = 65$Mbps.

From the Figures 6.3(a) to 6.10(c), we notice the following:

1. The EGTCA setup always outperforms the E-Greedy setup for the eLAA network and the WiFi with DCA ability network. This is due to the fact that the objective function used in EGTCA is more effective than the objective function of the Enhanced-Greedy CAM.

2. The EGTCA and E-Greedy have close performances with tremendous outperformance compared to the Greedy and E-Random CAMs in the eLAA network and the WiFi with DCA ability network. This is due to the fact that the EGTCA and Enhanced-Greedy CAMs possess all of three strength aspects while the Greedy and Enhanced-Random lack all or some of these aspects as mentioned previously.

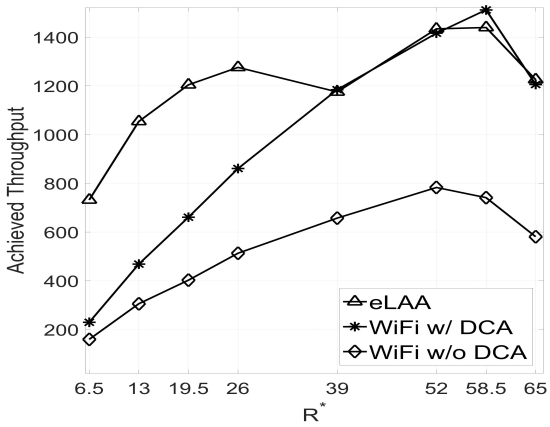
3. The eLAA performance is always better than the performance of both WiFi networks. Regarding the WiFi network with no DCA ability, the reason is very clear, since the eLAA uses better CAM than the Random CAM used by the this WiFi network. Regarding the WiFi network with DCA ability, even though this WiFi network uses the same CAM as the eLAA network but the outperformance of the eLAA is due to the fact that the eNB according to the LTE standard can serve multiple users simultaneously, while the WiFi AP can serve the users one at a time. This is very clear when the

required data rates have low values. For example, if there are three eNB users and three AP users. Let us assume each user requires 6.5Mbps data rate and that the eNB and the AP can support all of their three corresponding users on the chosen channel. Let us also assume that each user transmission requires 1 second to be done, then the total throughput achieved by the eNB cell is $\frac{(6.5+6.5+6.5)\text{Mbit}}{1\text{s}} = 19.5\text{Mbps}$, while total throughput achieved by the AP cell is $\frac{(6.5+6.5+6.5)\text{Mbit}}{3\text{s}} = 6.5\text{Mbps}$. This example shows clearly why the eLAA outperforms the WiFi network. And for this reason, it is expected for the MulteFire networks to be a fierce competitor to WiFi. For this reason, it is also expected for the LWA networks to have the same performance as WiFi networks as was stated previously in Section 6.1.

4. We notice that the WiFi network with DCA ability that uses the Greedy CAM have a throughput dip as the number of channels increases when R^* is low. This dip vanishes and turns into a "temporary saturation" as R^* increases. We also notice that this dip does not exist in the eLAA network at any R^* value. As it was explained before, this "temporary saturation" is one form of the "piling effect". This "temporary saturation" turns into a "temporary dip" in the case of WiFi when R^* is low. This is due to the fact that WiFi AP serves the users one-by-one instead of concurrently. This means if the eLAA network was able to achieve the "temporary saturation" level at low R^*, then the WiFi network should have this dip at low R^* because it takes more time to serve its users compared to the eLAA as it is clear form the previous point. In other words, this dip is due to the "piling effect" and the fact that WiFi serves the users one-by-one instead of concurrently. The eLAA was able to maintain the "temporary saturation" at low R^* values, because it can serve multiple users at the same time which provides higher throughput. We also notice that as R^* increases, the "temporary dip" changes to "temporary saturation" in the WiFi network, and "temporary saturation" changes to throughput increase in the eLAA network. This also can be explained in the same way by using the idea of the "piling effect" and the method of serving the users (i.e.,

one-by-one or simultaneously).

5. If we look at the achieved throughput curves of any network, we can notice that the throughput increases as R^* increases up to a point then it starts decreasing. This is expected since as R^* increases the throughput should increase if the network can support this increase in R^*. But as R^* keeps increasing, the network will no longer be able to keep up with this increase, hence a throughput decrease is expected to happen. Figure 6.11(a) shows this behavior more clearly at $c = 25$ channels in the EGTCA setup. Other setups have similar behavior.



(a) EGTCA Throughput (Mbps) Vs R^* at $c = 25$ channels.

In Figure 6.11(a), we notice two dips in the eLAA throughput as R^* increases. The first dip can be explained by the fact that eLAA is unable to keep serving multiple users all asking for high data rates at the same time, while on the other hand the WiFi can serve these users on different time slots, hence, the first dip does not exist in the WiFi throughput curves. The second dip can be explained as before (i.e., due to the network inability to keep up with the increase in R^*).

6. Regarding the comparison between the Enhanced-Random and Random CAMs, we can see in the figures from 6.3(a) to 6.10(c) that the eLAA using the Enhanced-Random

outperforms the WiFi with Enhanced-Random CAM network and the WiFi with Random CAM network. This is because the eNB always tries to maximize the number of served users on any given channel. We also notice that the WiFi with Enhanced-Random CAM network outperforms the WiFi with Random CAM network. This because the Enhanced-Random keeps changing the operating frequency dynamically (i.e., per AW). On the other hand, the Random CAM remains on the chosen channel for a very long period of time.

6.5 Future Work

As a future work, it is important to improve the work proposed in this dissertation by

1. Combining the work in the first four chapters and the work in Chapters 5 and 6 to study the effect of imperfect detection on the proposed CAMs' performance.

2. Developing the existing LTE and WiFi MAC protocols to accommodate the proposed CAMs.

6.6 Conclusion

In this chapter, the interactions between three types of networks coexisting on the same frequency band are studied. These networks are the LTE-eLAA, a future proposal for WiFi networks that have the ability to dynamically assign their operating channels (i.e., WiFi with DCA ability), and the currently used WiFi networks that lack the DCA ability. These networks also coexist with out of system devices (OSDs) that use the same frequency band and act as sources of interference to the eLAA and WiFi networks.

Two channel assignment mechanisms (CAMs) are proposed, namely: EGTCA and Enhanced-Greedy. The proposed CAMs are an enhancement on the already existing CAMs namely: GTCA and Greedy CAMs, respectively. We found that a sound coexistence CAM must possess three aspects (called the strength aspects): 1) credible objective function, 2) the

ability to achieve the objective function on the chosen channel, and 3) the ability to account for the surrounding current network activities. The results show a tremendous improvement in the throughput and fairness values using the proposed CAMs compared to currently being used CAMs in WiFi networks (i.e., the Greedy and Random CAMs). Our results show that EGTCA has a better performance compared to the Enhanced-Greedy since the objective function in EGTCA is more effective in improving the throughput.

This chapter also provides the results and the explanation for the following findings:

- eLAA networks outperform (in terms of throughput and fairness) the WiFi networks with DCA ability when the users have uniformly distributed data rate requirements from the set of the data rate allowed by the system.

- eLAA networks outperform (in terms of throughput and fairness) the WiFi networks with DCA ability when the users have tendency for low data rate requirements from the set of the data rates allowed by the system.

- eLAA networks are outperformed by the WiFi networks with DCA ability (in terms of throughput and fairness) when the users have tendency for high data rate requirements from the set of the data rates allowed by the system.

- The results show that the coexistence on the unlicensed band is possible and can help in alleviating the congestion happening on the licensed band.

- Based on these findings, we are in the favor of considering the coexistence between LTE and WiFi on the unlicensed band as one of the main features of the next generation of broadband communication (i.e., 5^{th} generation or 5G).

CHAPTER 7

Conclusion

7.1 Conclusion

This dissertation proposes new frameworks for LTE and WiFi networks to be able to coexist on the unlicensed band. These frameworks target the MAC and PHY layers.

In the PHY layer, we present multiple efficient spectrum sensing and data fusion techniques that take into consideration the correlation among the sensed signals. The main objective of these techniques is to design the decision rules at the sensors and the global fusion rule at the fusion center (FC) in order to minimize the probability of error at the FC.

In the MAC layer, the following challenge is faced in this dissertation: choosing the channel that achieves the best user experience, while, providing fairness among the users in the same system and fairness among users of different systems. To address this challenge, this dissertation presents a new channel assignment mechanism for both LTE and WiFi. The main idea of this channel assignment is to choose the channel that is just good enough for the required transmission by the cell (whether LTE or WiFi). This idea is very effective in leaving the overqualified channels for other cells that may need them. Hence, high throughput and fairness can be achieved simultaneously. This channel assignment was tested on a WiFi-only system to achieve a better coexistence between the WiFi cells. The results show higher performance compared to both greedy and random channel assignment mechanisms. An IEEE 802.11-based WiFi MAC protocol is also presented to handle the control packets exchange needed for the proposed channel assignment to work as intended.

The same channel assignment is also added to the LTE system. The performance was tested under the coexistence with two types of WiFi networks. The first WiFi type is a

network that uses the proposed channel assignment while the second type is a WiFi network that uses the greedy and random channel assignment as currently used in off-the-shelf WiFi access points.

The results show that by using the proposed channel assignment a noticeable increase in throughput and fairness values can be achieved compared to the greedy and random channel assignments.

The main conclusion from this work was: the coexistence between LTE and WiFi on the unlicensed band is possible using the proposed frameworks on the PHY and MAC layers.

7.2 List of publications

This section lists all the papers that have been published out of this dissertation's content and up to the date of its publication.

Journals:

- H. Kasasbeh, R. Viswanathan, and L. Cao, "Noise Correlation Effect on Detection: Signals in Equicorrelated or Autoregressive(1) Gaussian," IEEE Signal Processing Letters, pp. 1078-1082, July 2017.
- H. Kasasbeh, L. Cao, and R. Viswanathan, "Soft-Decision Based Distributed Detection Systems with Correlated Sensing Channels", IEEE Transactions on Aerospace and Electronic Systems. 2018.

Conferences:

- H. Kasasbeh, L. Cao, and R. Viswanathan, "Hard Decision Based Distributed Detection in Multi-Sensor System over Noise Correlated Sensing Channels," 50^{th} Annual Conference on Information Sciences and Systems (CISS), NJ, USA, 2016.
- H. Kasasbeh, F. Wang, L. Cao, and R. Viswanathan, "Generous Throughput Oriented Channel Assignment for Infra-structured WiFi Networks", Proc. IEEE WCNC 2017, San Francisco, USA.

- H. Kasasbeh, L. Cao, and R. Viswanathan, "Soft Decision Based Distributed Detection over Correlated Sensing Channels", IEEE CISS 2017, Baltimore, USA.

- H. Kasasbeh, S. Challa, L. Cao, and R. Viswanathan, "Decentralized Detection of Signals with Joint Clayton Copula Distribution Model", IEEE WiSEE 2018, Huntsville, USA.

Workshop posters:

- H. Kasasbeh, L. Cao, and R. Viswanathan, "Hard Decision Based Distributed Detection in Multi- Sensor System over Noise Correlated Sensing Channels," Broadband Wireless Access & Application Center (BWAC), Oxford, MS, USA, November 2015.

- H. Kasasbeh, L. Cao, and R. Viswanathan, "Soft Decision Based Distributed Detection Over Correlated Sensing Channels," Broadband Wireless Access & Application Center (BWAC), Washington DC, USA, November 2016.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] 802.11a, I. S. (1999), IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part II: Wireless Medium Access Control (MAC) and Physical Layer (phy) specifications: High Speed Physical Layer in the 5 GHz band, *IEEE Std 802.11a-1999*.

[2] 802.11b, I. S. (2001), IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirement. Part II: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 2: Higher-Speed Physical Layer (PHY) Extension in the 2.4 GHz Band - Corrigendum 1, *IEEE Std 802.11b-1999/Cor 1-2001*.

[3] 802.11g, I. S. (2003), IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part II: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, *IEEE Std 802.11g-2003*, pp. i–67.

[4] 802.11n, I. S. (2009), IEEE Standard for Information Technology– Local and Metropolitan Area Networks– Specific requirements– Part II: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput, *IEEE Std 802.11n-2009*, pp. 1–565.

[5] Abdi, Y., and T. Ristaniemi (2014), Joint local quantization and linear cooperation in spectrum sensing for cognitive radio networks, *Signal Processing, IEEE Transactions on*, 62(17), 4349–4362.

[6] Abraham, N., P. P. E. Winston, and M. Vadivel (2014), Adaptive channel allocation algorithm for wifi networks, in *2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2014]*, pp. 1307–1311.

[7] Ahsant, B., R. Viswanathan, S. Jeyaratnam, and S. K. Jayaweera (2012), New results on large sample performance of counting rules, in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pp. 882–885, IEEE.

[8] Akbar, A., M. Jaseemuddin, X. Fernando, and W. Farjow (2015), Energy-efficient scheduled directional medium access control protocol for wireless sensor networks, in *Communications (ICC), 2015 IEEE International Conference on*, pp. 6258–6264.

[9] Akl, R., and A. Arepally (2007), Dynamic channel assignment in ieee 802.11 networks, in *Portable Information Devices, 2007. PORTABLE07. IEEE International Conference on*, pp. 1–5.

[10] Al-Dulaimi, A., S. Al-Rubaye, Q. Ni, and E. Sousa (2015), 5G communications race: Pursuit of more capacity triggers lte in unlicensed band, *IEEE vehicular technology magazine*, 10(1), 43–51.

[11] Alvi, S., M. Younis, M. Imran, F. Amin, and M. Guizani (2015), A near-optimal llr based cooperative spectrum sensing scheme for crahns, *Wireless Communications, IEEE Transactions on*, 14(7), 3877–3887.

[12] Baid, A., and D. Raychaudhuri (2015), Understanding channel selection dynamics in dense Wi-Fi networks, *IEEE Communications Magazine*, 53(1), 110–117.

[13] Baidas, M. W., A. S. Ibrahim, K. G. Seddik, and K. R. Liu (2009), On the impact of correlation on distributed detection in wireless sensor networks with relays deployment, in *2009 IEEE International Conference on Communications*, pp. 1–6, IEEE.

[14] Balbi, H., N. Fernandes, F. Souza, R. Carrano, C. Albuquerque, D. Muchaluat-Saade, and L. Magalhaes (2012), Centralized channel allocation algorithm for IEEE 802.11 networks, in *Global Information Infrastructure and Networking Symposium (GIIS)*, pp. 1–7.

[15] Beaulieu, N., and C. Leung (1986), Optimal detection of hard-limited data signals in different noise environments, *Communications, IEEE Transactions on*, 34(6), 619–622.

[16] Berkelaar, M., K. Eikland, P. Notebaert, et al. (2004), lpsolve: Open source (mixed-integer) linear programming system, *Eindhoven U. of Technology*.

[17] Boyd, J. P. (2014), *Solving Transcendental Equations: The Chebyshev Polynomial Proxy and Other Numerical Rootfinders, Perturbation Series, and Oracles*, SIAM.

[18] Cao, L., and R. Viswanathan (2014), Divergence-based soft decision for error resilient decentralized signal detection, *IEEE Transactions on Signal Processing*, 62(19), 5095–5106.

[19] Chamberland, J. F., and V. V. Veeravalli (2003), Decentralized detection in sensor networks, *IEEE Transactions on Signal Processing*, 51(2), 407–416.

[20] Chamberland, J. F., and V. V. Veeravalli (2006), How dense should a sensor network be for detection with correlated observations?, *IEEE Transactions on Information Theory*, 52(11), 5099–5106.

[21] Chamberland, J. F., and V. V. Veeravalli (2007), Wireless sensors in distributed detection applications, *IEEE Signal Processing Magazine*, 24(3), 16–25.

[22] Chaudhari, S., J. Lunden, V. Koivunen, and H. V. Poor (2012), Cooperative sensing with imperfect reporting channels: Hard decisions or soft decisions?, *IEEE Transactions on Signal Processing*, 60(1), 18–28.

[23] Chen, C., R. Ratasuk, and A. Ghosh (2015), Downlink performance analysis of LTE and WiFi coexistence in unlicensed bands with a simple listen-before-talk scheme, in *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*, pp. 1–5, IEEE.

[24] Chen, H., B. Chen, and P. K. Varshney (2012), A new framework for distributed detection with conditionally dependent observations, *IEEE Transactions on Signal Processing*, 60(3), 1409–1419.

[25] Chen, P.-N., and A. Papamarcou (1995), Likelihood ratio partitions for distributed signal detection in correlated gaussian noise, in *Proceedings of 1995 IEEE International Symposium on Information Theory*, pp. 118–.

[26] Chepuri, S. P., and G. Leus (2016), Sparse sensing for distributed detection, *IEEE Transactions on Signal Processing*, 64(6), 1446–1460.

[27] Ciuonzo, D., and P. Salvo Rossi (2014), Decision fusion with unknown sensor detection probability, *IEEE Signal Processing Letters*, 21(2), 208–212.

[28] Ciuonzo, D., G. Papa, G. Romano, P. Salvo Rossi, and P. Willett (2013), One-bit decentralized detection with a rao test for multisensor fusion, *IEEE Signal Processing Letters*, 20(9), 861–864.

[29] Ciuonzo, D., A. De Maio, and P. Salvo Rossi (2015), A systematic framework for composite hypothesis testing of independent bernoulli trials, *IEEE Signal Processing Letters*, 22(9), 1249–1253.

[30] Ciuonzo, D., P. Salvo Rossi, and P. Willett (2017), Generalized rao test for decentralized detection of an uncooperative target, *IEEE Signal Processing Letters*, 24(5), 678–682.

[31] Decker, E., V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves (2008), The multichannel flow-aware medium access control protocol for wireless sensor networks, in *Per-*

sonal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on, pp. 1–5.

[32] Drakopoulos, E., and C.-C. Lee (1991), Optimum multisensor fusion of correlated local decisions, *IEEE Transactions on Aerospace and Electronic Systems*, 27(4), 593–606.

[33] Elwekeil, M., M. Alghoniemy, O. Muta, A. Abdel-Rahman, H. Furukawa, and H. Gacanin (2014), A maxmin model for solving channel assignment problem in ieee 802.11 networks, in *Signal Processing Conference (EUSIPCO), Proceedings of the 22nd European*, pp. 146–150.

[34] Fu, Z., W. Xu, Z. Feng, X. Lin, and J. Lin (2017), Throughput analysis of lte-licensed-assisted access networks with imperfect spectrum sensing, in *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*, pp. 1–6, IEEE.

[35] Gazis, V., K. Sasloglou, N. Frangiadakis, P. Kikiras, A. Merentitis, K. Mathioudakis, and G. Mazarakis (2013), Cooperative communication in channel assignment strategies for ieee 802.11k wlan systems, in *Personal Indoor and Mobile Radio Communications (PIMRC), IEEE 24th International Symposium on*, pp. 1924–1929.

[36] Geraniotis, E., and Y. A. Chau (1988), Distributed detection of weak signals from multiple sensors with correlated observations, in *Decision and Control, 1988., Proceedings of the 27th IEEE Conference on*, pp. 2501–2506, IEEE.

[37] Goldsmith, A. (2005), *Wireless communications*, Cambridge university press.

[38] Graybill, F. (1983), *Matrices with Applications in Statistics*, Wadsworth International Group, Belmont, CA.

[39] Haykin, S. (2005), Cognitive radio: brain-empowered wireless communications, *Selected Areas in Communications, IEEE Journal on*, 23(2), 201–220.

[40] Hoballah, I. Y., and P. K. Varshney (1989), Distributed bayesian signal detection, *IEEE Transactions on Information Theory*, 35(5), 995–1000.

[41] Holland, J. H. (1975), *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.*, U Michigan Press.

[42] Iyengar, S. G., P. K. Varshney, and T. Damarla (2011), A parametric copula-based framework for hypothesis testing using heterogeneous data, *IEEE Transactions on Signal Processing*, 59(5), 2308–2319.

[43] Jain, R. (1990), *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*, John Wiley & Sons.

[44] Kasasbeh, H., L. Cao, and R. Viswanathan (2016), Hard decision based distributed detection in multi-sensor system over noise correlated sensing channels, in *2016 Annual Conference on Information Science and Systems (CISS)*, pp. 280–285, IEEE.

[45] Kasasbeh, H., L. Cao, and R. Viswanathan (2017), Soft-decision based distributed detection over correlated sensing channels, in *Conference on Information Sciences and Systems CISS*, pp. 1–6, IEEE.

[46] Kasasbeh, H., R. Viswanathan, and L. Cao (2017), Noise correlation effect on detection: Signals in equicorrelated or autoregressive (1) gaussian, *IEEE Signal Processing Letters*.

[47] Kasasbeh, H., F. Wang, L. Cao, and R. Viswanathan (2017), Generous throughput oriented channel assignment for infra-structured wifi networks, in *Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, IEEE.

[48] Kasasbeh, H., L. Cao, and R. Viswanathan (2018), Soft-decision based distributed detection with correlated sensing channels, *IEEE Transactions on Aerospace and Electronic Systems*.

[49] Kay, S. M. (1998), *Fundamentals of Statistical Signal Processing Volume II Detection Theory*, Prentice Hall PTR, Upper Saddle River, New Jersey 07458.

[50] Khalid, L., and A. Anpalagan (2012), Cooperative sensing with correlated local decisions in cognitive radio networks, *IEEE Transactions on Vehicular Technology*, 61(2), 843–849.

[51] Kuhn, H. W. (1955), The hungarian method for the assignment problem, *Naval research logistics quarterly*, 2(1-2), 83–97.

[52] Li, W., and H. Dai (2007), Distributed detection in wireless sensor networks using a multiple access channel, *IEEE Transactions on Signal Processing*, 55(3), 822–833.

[53] Li, Y., Q. Li, and J. Zheng (2015), Performance analysis of licensed-assisted access using lte with adaptive carrier selection, in *Globecom Workshops (GC Wkshps), 2015 IEEE*, pp. 1–6, IEEE.

[54] Li, Y., X. Zhang, and K. L. Yeung (2015), Dli: A dynamic listen interval scheme for infrastructure-based ieee 802.11 wlans, in *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2015 IEEE 26th Annual International Symposium on*, pp. 1206–1210.

[55] Lim, T. H., W. S. Jeon, and D. G. Jeong (2016), Centralized channel allocation scheme in densely deployed 802.11 wireless lans, in *2016 18th International Conference on Advanced Communication Technology (ICACT)*, pp. 249–253.

[56] Lima, M. P., T. B. Rodrigues, R. F. Alexandre, R. H. C. Takahashi, and E. G. Carrano (2014), Using evolutionary algorithms for channel assignment in 802.11 networks, in *Computational Intelligence for Communication Systems and Networks (CIComms), 2014 IEEE Symposium on*, pp. 1–8.

[57] Lin, X., and R. S. Blum (1998), Numerical solutions for optimum distributed detection of known signals in dependent t-distributed noise-the two sensor problem, in *Signals, Systems amp; Computers, 1998. Conference Record of the Thirty-Second Asilomar Conference on*, vol. 1, pp. 613–617 vol.1.

[58] Lin, Y., and H. Chen (2015), Distributed detection performance under dependent observations and nonideal channels, *IEEE Sensors Journal*, 15(2), 715–722.

[59] Ling, J., D. Lopez-Perez, and M. R. Khawer (2017), Practical lte and wifi coexistence techniques beyond lbt, *IEEE Communications Magazine*, 55(10), 127–133.

[60] Liu, C.-H., and H.-C. Tsai (2017), On the limits of coexisting coverage and capacity in multi-rat heterogeneous networks, *IEEE Transactions on Wireless Communications*, 16(5), 3086–3101.

[61] Ma, Y., S. Dehnie, and V. D. Chakravarthy (2015), On the near-optimality of training-based glrt spectrum sensing, *IEEE Transactions on Wireless Communications*, 14(9), 4894–4906.

[62] Matsumura, Y., S. Kumagai, T. Obara, T. Yamamoto, and F. Adachi (2012), Channel segregation based dynamic channel assignment for wlan, in *Communication Systems (ICCS), 2012 IEEE International Conference on*, pp. 463–467.

[63] Poor, H. V. (2013), *An introduction to signal detection and estimation*, Springer Science & Business Media.

[64] Quan, Z., S. Cui, and A. H. Sayed (2008), Optimal linear cooperation for spectrum sensing in cognitive radio networks, *Selected Topics in Signal Processing, IEEE Journal of*, 2(1), 28–40.

[65] Srinath, M. D., P. Rajasekaran, and R. Viswanathan (1995), *Introduction to statistical signal processing with applications*, Prentice-Hall, Inc.

[66] Srinivasan, R. (1986), Distributed radar detection theory, *Communications, Radar and Signal Processing, IEE Proceedings F*, 133(1), 55–60.

[67] Tang, Z.-B., K. R. Pattipati, and D. L. Kleinman (1991), An algorithm for determining the decision thresholds in a distributed detection problem, *Systems, Man and Cybernetics, IEEE Transactions on*, 21(1), 231–237.

[68] Temma, K., Y. Matsumura, and F. Adachi (2013), Multi-channel access in wireless networks using interference-aware channel segregation based dynamic channel assignment, in *Wireless Communications & Signal Processing (WCSP), International Conference on*, pp. 1–5, IEEE.

[69] Tsinos, C. G., F. Foukalas, and T. A. Tsiftsis (2017), Resource allocation for licensed/unlicensed carrier aggregation mimo systems, *IEEE Transactions on Communications*, 65(9), 3765–3779.

[70] Tsitsiklis, J. N., et al. (1993), Decentralized detection, *Advances in Statistical Signal Processing*, 2(2), 297–344.

[71] Unnikrishnan, J., and V. V. Veeravalli (2007), Decentralized detection with correlated observations, in *2007 Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*, pp. 381–385.

[72] Van Trees, H. L. (2004), *Detection, estimation, and modulation theory*, John Wiley & Sons.

[73] Varshney, P. K. (2012), *Distributed detection and data fusion*, Springer Science & Business Media.

[74] Viswanathan, R., and B. Ahsant (2012), A review of sensing and distributed detection algorithms for cognitive radio systems., *International Journal on Smart Sensing & Intelligent Systems*, 5(1).

[75] Viswanathan, R., and P. K. Varshney (1997), Distributed detection with multiple sensors Part I. fundamentals, *Proceedings of the IEEE*, 85(1), 54–63.

[76] Willett, P., P. Swaszek, and R. Blum (2000), The good, bad and ugly: distributed detection of a known signal in dependent gaussian noise, *IEEE Transactions on Signal Processing*, 48(12), 3266–3279.

[77] Xiao, J., and J. Zheng (2016), An adaptive channel access mechanism for lte-u and wifi coexistence in an unlicensed spectrum, in *Communications (ICC), 2016 IEEE International Conference on*, pp. 1–6, IEEE.

[78] Yakoubsohn, J.-C. (2005), Numerical analysis of a bisection-exclusion method to find zeros of univariate analytic functions, *Journal of Complexity*, 21(5), 652–690.

[79] Ying, X., and I. N. Katz (1989), A simple reliable solver for all the roots of a nonlinear function in a given domain, *Computing*, 41(4), 317–333.

[80] Yue, X., C. F. Wong, and S. H. G. Chan (2011), Cacao: Distributed client-assisted channel assignment optimization for uncoordinated wlans, *IEEE Transactions on Parallel and Distributed Systems*, 22(9), 1433–1440.

[81] Zhao, W., Z. Fadlullah, H. Nishiyama, N. Kato, and K. Hamaguchi (2014), On joint optimal placement of access points and partially overlapping channel assignment for wireless networks, in *Global Communications Conference (GLOBECOM)*, pp. 4922–4927, IEEE.

VITA

I received my bachelor's and master's degrees in telecommunications and wireless communications engineering from Yarmouk University, Irbid, Jordan, in 2012, and 2014, respectively.

My research interests include the areas of spectrum sensing, distributed detection, and MAC protocol and resource allocation mechanism designs in wireless networks.

In April 2019, I started working as an RF research engineer for AeroDefense LLC. in New Jersey, USA. AeroDefense is a company that provides detection and localization systems of malicious drones.